

## Localización geográfica de buques

**Miguel Ángel Malpartida Jiménez**

Grado de Ingeniería informática

**Consultor: María Isabel March Hermo**

Fecha Entrega:01/2020



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## Copyright

© (Miguel Ángel Malpartida Jiménez)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	Localización geográfica de buques
<b>Nombre del autor:</b>	Miguel Ángel Malpartida Jiménez
<b>Nombre del consultor:</b>	María Isabel March Hermo
<b>Fecha de entrega (mm/aaaa):</b>	01/2020
<b>Área del Trabajo Final:</b>	Redes de computadores
<b>Titulación:</b>	<i>Grado de ingeniería informática</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b>	
<p>En la actualidad, el transporte marítimo experimenta un continuo incremento a nivel mundial. Los hábitos de compra a través de Internet unido a un comercio global hacen que, detrás de esta, exista un barco transportando nuestra mercancía.</p> <p>De igual forma, vivimos en un constante avance en las telecomunicaciones. La demanda de una comunicación rápida, eficiente y de mejor calidad hace que los sistemas se encuentren inmersos en un continuo y permanente avance.</p> <p>En el ámbito marítimo, con el propósito de mejorar la seguridad marítima se ideó el sistema de identificación automática AIS. Este sistema permite el intercambio de datos de navegación entre las embarcaciones y centros de control de tráfico marítimo equipados con dicho sistema.</p> <p>Este proyecto consiste en la creación de una aplicación de escritorio la cual permite monitorizar, mediante un mapa geográfico, los datos de identificación y posicionamiento que los buques transmiten a través del sistema AIS.</p> <p>Para lograr este objetivo, la aplicación procesa, al mismo tiempo en que se está recibiendo, la información enviada por un receptor AIS a través de una red de comunicaciones a la cual están conectados.</p> <p>El usuario obtendrá información relativa a cada embarcación y podrá visualizar gráficamente un escenario de tráfico marítimo formado por los buques que se encuentren dentro de la zona de cobertura.</p> <p>Mediante el análisis y diseño se han definido los requisitos que la aplicación debe satisfacer, así como la interacción con el usuario. El resultado final pasa por la implementación de una aplicación muy automatizada, intuitiva y fácil de usar.</p>	

**Abstract (in English, 250 words or less):**

Actually, maritime transport is experiencing a continuous increase worldwide. The habits of buying through the Internet together with a global trade mean that, behind it, there is a vessel carrying our merchandise.

Similarly, we live in a constant advance in telecommunications. The demand for fast, efficient and better quality communication makes the systems immersed in a continuous and permanent advance.

Related to the maritime field, the AIS automatic identification system was designed to improve maritime safety. This system allows the exchange of navigation data between vessels and maritime traffic control centers equipped with that system.

This project consists in the develop of a desktop application which allows visualizing, by means of a geographical map, the identification and positioning data that vessels transmit through the AIS system.

To achieve this objective, the application processes, at the same time it is being received, the information sent by an AIS receiver through a communications network which they are connected.

The user will obtain information related to each vessel and can graphically visualize a maritime traffic scenario formed by the ships that are within coverage area.

Through the analysis and design, the requirements that the application must satisfy, as well as the interaction with the user, have been defined. The final result is the implementation of a highly automated, intuitive and easy to use application.

**Palabras clave (entre 4 y 8):**

AIS, Receptor, Socket, NMEA, OMI

## Índice

1. Introducción.....	1
1.1 Contexto y justificación del Trabajo.....	1
1.2 Objetivos del Trabajo.....	2
1.3 Enfoque y método seguido.....	3
1.4 Planificación del Trabajo .....	4
1.5 Breve resumen de productos obtenidos .....	8
1.6 Breve descripción de los otros capítulos de la memoria.....	9
2. Conceptos teóricos.....	10
2.1 Sistema AIS.....	10
2.2 Protocolos implicados.....	12
2.3 Interpretación de las sentencias.....	14
3. Análisis y diseño de la aplicación .....	20
3.1 Definición de los requisitos:.....	20
3.2 Diagrama de clases.....	21
4. Implementación .....	27
4.1 Conexión con el receptor.....	29
4.2 Procesamiento de la información .....	31
4.3 Decodificación de la información .....	33
4.4 Representación de la información .....	36
5. Pruebas .....	39
6. Manual de usuario .....	43
7. Conclusiones.....	50
8. Glosario .....	51
9. Bibliografía .....	53

## Lista de figuras

<i>Ilustración 1 Imagen del producto obtenido</i>	8
<i>Ilustración 2 Transmisión de la información con división en el tiempo</i>	11
<i>Ilustración 3 Uso típico en los equipos de navegación</i>	13
<i>Ilustración 4 Diagrama de clases</i>	21
<i>Ilustración 5 Casos de usos</i>	22
<i>Ilustración 6 Mapa geográfico</i>	27
<i>Ilustración 7 Comportamiento del sistema</i>	28
<i>Ilustración 8 Cuadro resumen de las operaciones</i>	29
<i>Ilustración 9 Crear cliente socket</i>	30
<i>Ilustración 10 Dirección IP y Puerto</i>	30
<i>Ilustración 11 Envía petición</i>	30
<i>Ilustración 12 Recibir datos</i>	30
<i>Ilustración 13 Cerrar socket</i>	31
<i>Ilustración 14 Codificación array de bytes a string</i>	31
<i>Ilustración 15 Separación cadena de texto por línea</i>	31
<i>Ilustración 16 Array mayor de un elemento y por tanto obtención de la línea</i>	32
<i>Ilustración 17 Función que devuelve un BitArray</i>	33
<i>Ilustración 18 Bitarray</i>	33
<i>Ilustración 19 Desplazamiento de bit a la izquierda</i>	34
<i>Ilustración 20 Pasa de bit a entero con signo.</i>	34
<i>Ilustración 21 Pasa de bit a ASCII</i>	35
<i>Ilustración 22 Ejemplo se sentencias AIVDM, AIVDO y su decodificación:</i>	35
<i>Ilustración 23 Gmap.net</i>	36
<i>Ilustración 24 Parámetros de inicialización del mapa.</i>	37
<i>Ilustración 25 Procedimiento que crea o actualiza los marcadores</i>	37
<i>Ilustración 26 Borrado de buques</i>	38
<i>Ilustración 27 Función para rotar una imagen en función del rumbo.</i>	38
<i>Ilustración 28 Visualización sentencias usando Telnet</i>	39
<i>Ilustración 29 Si no introducimos IP</i>	40
<i>Ilustración 30 Si no introducimos puerto</i>	40
<i>Ilustración 31 Visualización de las sentencias</i>	41
<i>Ilustración 32 Lista con datos procesados</i>	41
<i>Ilustración 33 Aplicación en operación</i>	42
<i>Ilustración 34 Archivo a descomprimir</i>	43
<i>Ilustración 35 Archivo setup.exe</i>	43
<i>Ilustración 36 Carpeta de instalación</i>	43
<i>Ilustración 37 Introducimos dominio/IP y puerto</i>	44
<i>Ilustración 38 Resolver nombre de dominio</i>	45
<i>Ilustración 39 Numero de blancos, Conectar y Desconectar Socket</i>	45
<i>Ilustración 40 Sistema en operación</i>	46
<i>Ilustración 41 Información del buque</i>	47
<i>Ilustración 42 Campo de control para el borrado con valor true.</i>	47
<i>Ilustración 43 Visualización de las sentencias</i>	48
<i>Ilustración 44 Sentencias decodificadas</i>	48
<i>Ilustración 45 Lista con los buques a borrar</i>	49
<i>Ilustración 46 Lista con marcadores antes y después de borrar</i>	49

# 1. Introducción

## 1.1 Contexto y justificación del Trabajo

Después de años de experiencia laboral trabajando con herramientas que ayudan a la gestión de tráfico y seguridad marítima, vi en el proyecto final de grado, la oportunidad de poder estudiar y profundizar el conocimiento en el sistema AIS. A lo largo de este tiempo he utilizado aplicaciones desarrolladas por otros fabricantes y conforme avanzaban mis estudios en la UOC sentía gran curiosidad por cómo enfrentar un proyecto, aunque de menor dimensión, de similares características.

Con el paso del tiempo he ido viendo la evolución de este sistema y como los receptores actuales del sistema AIS, nos permiten intercambiar los mensajes de posicionamiento recibidos por los buques a un PC local mediante una interfaz Ethernet, en algunos casos, cableada y en otros utilizando una conexión inalámbrica. Estos dispositivos nos permiten utilizar Sockets para establecer una conexión usando una red de comunicaciones y fue este punto, el que me impulso a proponer este proyecto en el área de redes de computadores.

Existen multitud de equipos electrónicos que utilizan este tipo de comunicación para intercambiar información, ya que son muy eficaces a la hora de enviar y recibir un gran volumen de información.

El sistema AIS, es un elemento importante para mejorar la seguridad marítima. La gran utilidad de este sistema hace que muchas embarcaciones, aunque no tengan la obligatoriedad de llevarlo, quieran disponer de tecnologías que les permita mejorar la seguridad durante la navegación debido a que les garantiza su visibilidad desde otros barcos equipados con este sistema.

La posibilidad de trabajar con estos tipos de dispositivos mediante redes y protocolos de comunicaciones, unido a la concienciación y conocimiento que representa la seguridad marítima, hizo que despertara un gran interés para implementar una aplicación que recogiera información de un receptor AIS a través de una red de área local (LAN) y una vez tratada la información, poder mostrarla en un mapa geográfico. El resultado pasa por presentar un escenario de tráfico marítimo compuesto únicamente por las embarcaciones que incorporan este tipo de tecnología.

Otros de los motivos que llamaba mi atención, una vez obtenida la primera aproximación de lo que había que desarrollar, fue el tratamiento de la información. El trabajar procesando los datos recibidos mediante protocolos nuevos y como proporcionar una solución, suponía un reto nuevo para mí en el desarrollo de aplicaciones.

## 1.2 Objetivos del Trabajo

El objetivo principal es la implementación de una aplicación que represente un sistema capaz de monitorizar el tráfico marítimo en tiempo real.

Se pretende conseguir un producto el cual se comunique con un receptor AIS mediante una conexión Socket (cliente-servidor), y una vez conectado y tratada la información recibida, poder representarla en un mapa geográfico.

Mediante este sistema el usuario podrá monitorizar la información AIS a través de una interfaz totalmente funcional y fácil de usar. El operador dispondrá de la posibilidad de conexión a un receptor indicando una dirección IP y puerto.

Al mismo tiempo, el usuario podrá visualizar la siguiente información referente a cada buque:

- MMSI    identificativo del barco.
- Latitud.
- Longitud.
- Nombre del barco

Además, con el fin de que pueda resultar útil para los usuarios que la utilicen, la aplicación deberá conseguir el siguiente comportamiento:

- La información se mostrará al mismo tiempo en que se está recibiendo.
- El sistema se mostrará estable ante las interacciones del usuario. Se proporcionará una retroalimentación ante los posibles fallos que pueda ocasionar.
- La aplicación será capaz de procesar secuencialmente un número de blancos con el margen suficiente respecto al número de barcos que se encuentren en cobertura de un solo receptor.
- Se ha de tener en cuenta el borrado de un barco cuando este último sale de la zona de cobertura.

Será una aplicación muy automatizada y fácil de usar.



### 1.3 Enfoque y método seguido

Teniendo en cuenta factores como el tiempo, y en una primera clasificación con un objetivo claro, pero de solución poco conocida, se ha optado por un enfoque que permita ir cambiando las ideas iniciales en función del avance del proyecto.

Se hizo un primer estudio de los protocolos necesarios y de herramientas similares de acceso libre las cuales ofrecían una idea de cómo tendría que ser la representación gráfica de la información.

Una vez completado el estudio y realizado las especificaciones técnicas, se dedicó un poco más de tiempo al análisis funcional y diseño del sistema mediante el cual, se obtuvieron los requisitos, el diseño a implementar y los casos de uso.

Concluido el diseño, se pasa a la fase de implementación, la cual se ha dividido en cuatro etapas fundamentales:

- Una primera tarea encargada de realizar la conexión con el receptor mediante un cliente Socket. Este apartado consiste en recibir continuamente datos de buques procedentes de un receptor AIS. Se ha implementado utilizando un cliente Socket asíncrono y utilizando un buffer en el que se van almacenando los bytes que se van recibiendo.
- Una segunda tarea la cual consiste en clasificar todo este flujo de información en una estructura que nos permite obtener los datos línea a línea. Una vez obtenida, se divide esta última en campos y es, a este nivel, el que usaremos para el tratamiento de la información.
- Una vez clasificada la información, se pasa a la tarea de decodificación de las sentencias. Durante esta tarea se extraerán los datos relevantes que permitan la representación de la información perteneciente a cada buque.
- Por último, se utiliza esta información para posteriormente representarla en un mapa geográfico. Este último permitirá visualizar, en una determinada zona de cobertura, un escenario de tráfico marítimo en tiempo real.

En cada fase se han realizado las siguientes pruebas:

- Para la conexión con el receptor, ha sido útil el uso de herramientas como telnet con la finalidad de comprobar que los datos se transmiten correctamente a través de la red. Posteriormente, una vez establecida la conexión Socket, se puede visualizar por consola las sentencias recibidas.

- Una vez recibida, la dificultad se centraba en separar el flujo de información en líneas y a su vez, cada una, en campos. Para ello ha sido fundamental el uso de arrays y métodos para trabajar con cadenas.
- La fase de decodificación ha resultado ser la más compleja. Para ello ha sido determinante la práctica sobre papel, y una vez entendido, pasar a la implementación.
- En la representación de la información, el uso de marcadores ha resultado ser la tarea principal. Las pruebas manuales como: crear, borrar y mover marcadores sobre la capa del mapa han sido fundamentales para lograr la posterior automatización de los mismos.

Se tuvo que cambiar el diseño cuando se detectó las limitaciones que existían, insertando y borrando datos, utilizando variables de tipo estructura. Finalmente se optó por la creación de una clase cuyas instancias representarían cada buque. Esta elección permitió el uso de colecciones dinámicas como arraylist.

Mediante este método, se ha logrado la consecución de soluciones parciales habiendo sido fácil la corrección e incluso cambios en el planteamiento tal y como se ha comentado anteriormente.

## 1.4 Planificación del Trabajo

En este apartado se detalla las tareas a realizar y una estimación del tiempo por semanas que podrán tomar cada una de las actividades.

### Tarea 1:

Desde el 30 de Septiembre hasta el 13 de Octubre. Duración 2 semanas.

Descripción: Estudio, investigación y recogida de la información necesaria para la correcta realización del proyecto.

- Sistema de Identificación Automática (AIS).
- Protocolo NMEA y decodificación de los mensajes AIVDM/AIVDO.
- Análisis, diseño y arquitectura del sistema.
- Conexión con el receptor.

Objetivos:

- Estudio del sistema AIS, conceptos generales.
- Estudio y análisis del protocolo NMEA
- Estudio y análisis de los protocolos y comprensión de las sentencias recibidas AIVDM/AIVDO.
- Estudio y análisis del diseño a utilizar.

Metas:

- Familiarización con el sistema de identificación automática de buques, su arquitectura y protocolos necesarios para el posterior procesamiento de la información obtenida mediante un receptor AIS.

**Tarea 2:**

Desde el 14 de Octubre hasta el 10 de Noviembre. Duración 4 semanas.

Descripción: Análisis funcional y definición de los requisitos del sistema.

Objetivos:

- Definir los requisitos funcionales de la aplicación.
- Definir el diseño a implementar.
- Definición de los objetivos.
- Definición de las estructuras de datos a tratar.

Metas:

- Obtención del diseño de la aplicación y la arquitectura general del sistema.

**Tarea 3:**

Desde el 11 de Noviembre hasta el 15 de Diciembre. Duración 5 semanas.

Descripción: Implementación de la aplicación.

Objetivos:

- Conexión mediante cliente Socket al receptor AIS el cual hará de servidor Socket enviando la información.
- Tratamiento del flujo de información de entrada.
- Decodificación de las sentencias.
- Instalación de los componentes necesarios para la posterior representación en mapa Geo-referenciado.
- Representación de los datos decodificados en mapa mediante marcadores.
- Representación de la información referente a cada marcador.

Metas:

- El objetivo será la lectura de la información procedente del receptor AIS, su posterior tratamiento y decodificación para finalmente poder presentarla en tiempo real mediante mapa geo-referenciado.

**Tarea 4:**

Desde 16 de Diciembre hasta el 22 de Diciembre. Duración 1 semana.

Descripción: Realización de pruebas, depuración del código y resolución de posibles problemas que puedan surgir.

Objetivos:

- Obtención de una aplicación que represente la información obtenida de un receptor AIS mediante Sockets.
- Representación de la información referente a cada buque.

Metas:

- Comprobación de un escenario en tiempo real del tráfico marítimo en la zona de cobertura del receptor.

### **Tarea 5:**

Desde 14 de Octubre hasta el 23 de Diciembre. 10 semanas

Descripción: Realización de la documentación del proyecto. Esta tarea se llevará a cabo en paralelo con las tareas 2, 3 y 4.

#### Objetivos:

- Finalización de la documentación del proyecto.

#### Metas:

- Memoria del proyecto.

### **Tarea 6:**

Desde 06 de Enero al 12 de Enero. Duración 1 semana.

Descripción: Preparación de una presentación la cual refleje un resumen de las partes fundamentales del proyecto.

#### Objetivos:

- Realización de una síntesis de la memoria del proyecto.
- Realización de la presentación.

#### Metas:

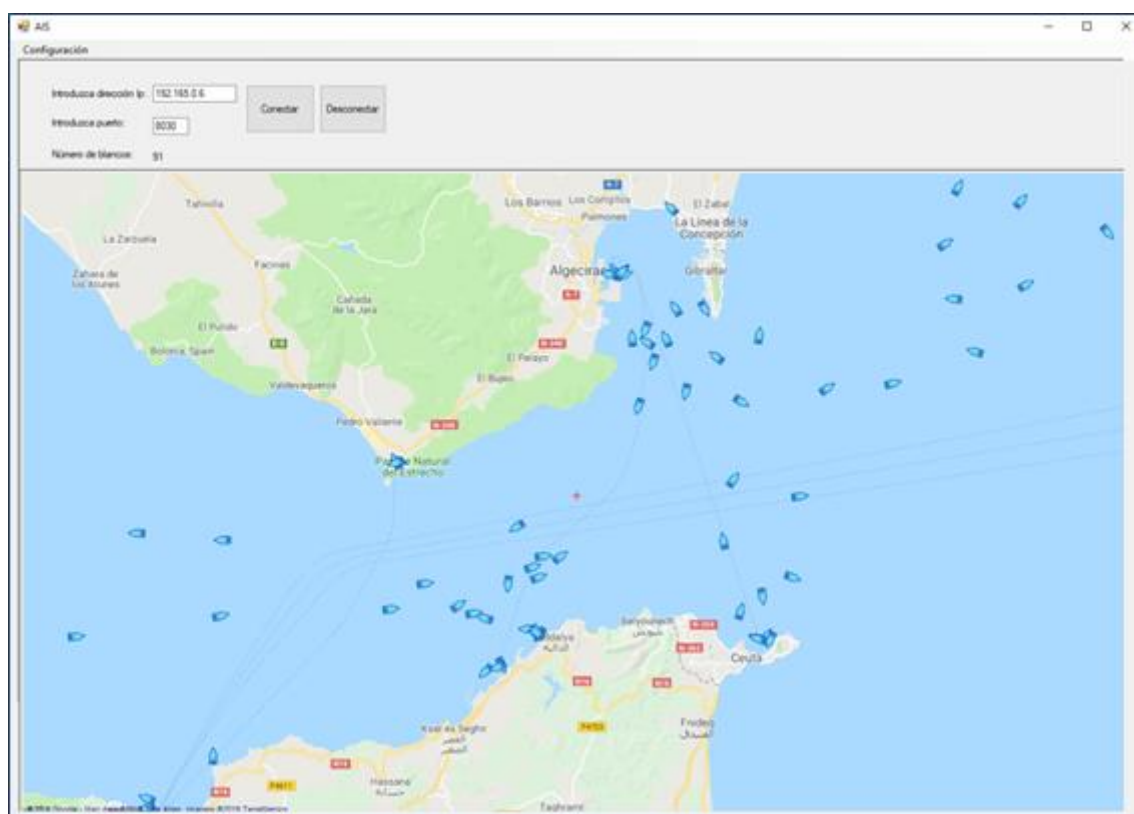
- Presentación del proyecto.

En principio, la tarea 4 y 5 se tenía prevista terminar para el 22 y 23 de Diciembre que coincidía con el inicio del periodo de entrega del plan docente. Una vez en la fase de implantación, se tuvieron que hacer algunos cambios en el diseño modificando el planteamiento inicial con lo que se ha optado por entregar la memoria y el producto el día 5 de Enero siguiendo el plan de entrega del TFG.

## 1.5 Breve resumen de productos obtenidos

El producto obtenido es una aplicación que presenta, mediante un mapa geográfico, el tráfico marítimo de buques en tiempo real. Dicha aplicación está desarrollada utilizando tecnología de cliente Windows Form usando el Framework .NET de Microsoft. El entorno de desarrollo utilizado es Microsoft Visual Studio 2019 en su versión gratuita.

El desarrollo consiste en presentar la información dinámica y estática que los buques van transmitiendo mediante el sistema AIS. Esta información es recibida y presentada en un mapa geo-localizado.



*Ilustración 1 Imagen del producto obtenido*

## **1.6 Breve descripción de los otros capítulos de la memoria**

A continuación, se expone una introducción de los capítulos específicos que componen la memoria:

En el capítulo dos “Conceptos teóricos” se describe el sistema AIS sus principales objetivos, clases, normas de aplicación y protocolos necesarios para la implementación de este proyecto. Se presenta el protocolo NMEA y la interpretación de las sentencias recibidas.

El siguiente capítulo “Análisis y diseño”, describe las necesidades y restricciones que afectan al proyecto en forma de requisitos funcionales y no funcionales. También, se presenta el diagrama de clases y los casos de usos.

El capítulo cuatro “Implementación”, comienza con una introducción describiendo el diseño a implementar y el comportamiento del sistema. Posteriormente, se describe los principales bloques que componen esta fase. Estos son: la conexión/comunicación con el receptor AIS, el procesamiento de la información, la decodificación de las sentencias recibidas y por último, se describe los componentes gráficos utilizados para la representación de la información.

En el capítulo cinco “Pruebas”, se recogen las pruebas realizadas donde se puede evaluar la calidad del producto y el correcto funcionamiento, así como los posibles fallos y en tal caso, la respuesta de la aplicación.

El capítulo seis “Manual de usuario” describe el proceso de instalación y puesta en marcha de la aplicación. También, se describe con detalle el uso que el usuario puede hacer con el sistema.

## 2. Conceptos teóricos

### 2.1 Sistema AIS

AIS es un sistema de identificación automática que nos permite conocer en tiempo real datos de identificación, posición y otras informaciones relevantes procedentes de los buques. Este sistema permite la comunicación entre las embarcaciones y estaciones de control de tráfico marítimo (VTS) de forma automática y con una gran disponibilidad.

El objetivo principal de este sistema es mejorar la seguridad de la navegación mediante la prevención de accidentes y facilitando el control del tráfico marítimo.

El sistema AIS fue adoptado por la Organización Marítima Internacional en el año 2002 y comenzó a operar en diciembre de 2004.

La regulación SOLAS 19 recoge la obligatoriedad del uso de este sistema en las siguientes embarcaciones:

- Buques de más 300 toneladas que participen en viajes internacionales.
- Buques de más de 500 toneladas que no participen en viajes internacionales.
- Todos los buques de pasajeros independientemente a su tamaño.

Además de esta regulación, cada Administración puede imponer su propia obligatoriedad en el uso de este sistema.

Existen dos tipos de AIS: sistema de clase A y sistema de clase B.

El de clase A es un sistema de uso obligatorio y se utiliza para complementar a los sistemas de navegación de un buque cumpliendo con las especificaciones SOLAS. Transmiten la información con una potencia de 12,5 vatios aportando una cobertura que se limita al rango de visión (aproximadamente 20-30 millas). Los transceptores AIS de clase A poseen un receptor DSC (156.525 MHz), GPS externo, rumbo e indicador de velocidad de giro. Además, pueden transmitir y recibir mensajes de texto relacionados con la seguridad.

Los de clase B es un sistema menos complejo, de uso no obligatorio, basado en transceptores más pequeños utilizados en embarcaciones de menor tamaño. Para evitar la saturación del ancho de banda disponible, la potencia de transmisión es de 2 vatios, limitando el rango de cobertura entre 5 y 10 millas. El receptor DSC y rumbo son opcionales. La transmisión de mensajes de texto relacionados con la seguridad también es opcional y está disponible solo cuando viene configurado.

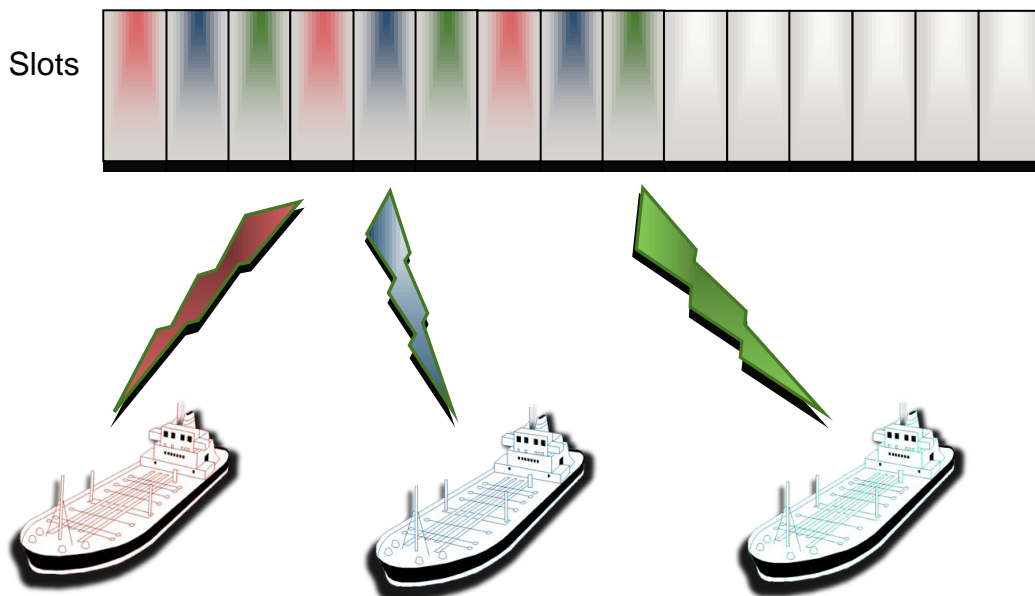


Existen embarcaciones como buques militares, embarcaciones pequeñas o náutica deportiva que no están obligados a llevar este sistema, con lo que es importante, recalcar que una pantalla de AIS no es un reflejo exacto de la realidad. Existe el riesgo de que el sistema no presente buques que realmente están navegando con el consiguiente peligro para la navegación.

La tecnología AIS integra un transceptor VHF con un sistema de posicionamiento GPS. Opera en dos canales de banda marina: AIS 1 87B (frecuencia 161.975) y AIS 2 88B (frecuencia 162.025).

Estos sistemas transmiten información como: posición, rumbo y velocidad para que otros buques o estaciones de tierra puedan conocer su posición exacta.

Los paquetes de datos se transmiten de forma sincronizada con una secuencia de tiempos predeterminada. Para que puedan funcionar de forma conjunta y evitar el solapamiento de la señal en el mismo tiempo, los transceptores utilizan la tecnología TDMA (Time Division Multiple Access). Mediante esta técnica de multiplexación, los datos se clasifican en unidades de información llamado Slots, alternos en el tiempo, proporcionando un acceso múltiple donde cada transceptor solicita un slot de tiempo para transmitir la información.



*Ilustración 2 Transmisión de la información con división en el tiempo*

### Periodicidad de actualización de la información:

La información es actualizada en función de los siguientes parámetros:

Información estática: Cada 6 minutos y cuando se solicite.

Información dinámica: Se actualiza según parámetros descritos en la tabla1:

Tabla 1

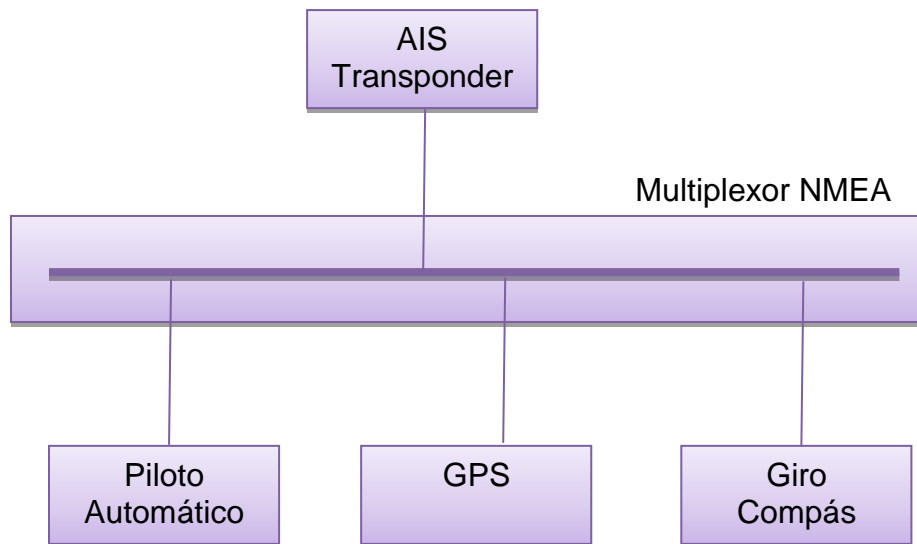
Tipo de barco	Intervalo
Barco anclado	3 minutos
Barco en movimiento de 0 a 14 nudos	12 segundos
Barco en movimiento de 0 a 14 con cambio de derrotero	4 segundos
Barco en movimiento de 14 a 23 nudos	6 segundos
Barco en movimiento de 14 a 23 nudos con cambio de derrotero	2 segundos
Barco en movimiento de más de 23 nudos	3 segundos
Barco en movimiento a más de 23 nudos con cambio de derrotero	2 segundos

## 2.2 Protocolos implicados

Protocolo NMEA 0183. Es una especificación que define la interfaz para la comunicación de equipos electrónicos marinos. Este estándar permite el envío de información a las computadoras y a otros equipos instalados en los buques.

La finalidad principal es permitir a los usuarios y fabricantes la capacidad de combinar hardware y software mediante normas que regulan la comunicación entre diferentes sistemas los cuales transmiten información a través de diferentes medios.

Es un protocolo que define los requerimientos de datos y velocidad de transmisión en formato serie. Mediante este protocolo tenemos equipos emitiendo y recibiendo datos entre varios dispositivos. Solo permite la transmisión de información simultánea por un dispositivo mientras que los demás equipos permanecen a la escucha. En el caso de tener más equipos transmitiendo información, se necesita un multiplexor de señales. Estos dispositivos permiten la convergencia de datos en un bus NMEA.



*Ilustración 3 Uso típico en los equipos de navegación*

Existen diferentes tipos de mensajes NMEA con distintas capacidades. Además, en la actualidad, los datos NMEA se pueden transmitir a través de diferentes tipos de interfaces de comunicación.

Los mensajes se transmiten de forma asíncrona y sin paridad. Existen 3 tipos de sentencias:

- Sentencias informativas. Estas sentencias tienen un máximo de 80 caracteres además del inicio de línea y el retorno de carro. Si no hay datos para transmitir hay que construir la sentencia con las comas sin ningún espacio entre ellas.
- Sentencias propietarias. Son utilizadas para que los fabricantes puedan construir sus propias sentencias.
- Sentencias interrogativas. Posibilita a otro dispositivo solicitar una sentencia informativa. Los primeros caracteres identifican al solicitante y los dos siguientes identifican al interrogado.

Los datos están contenidos dentro de una línea separados por comas. La información está compuesta por caracteres ASCII y normalmente se encuentran en una sentencia de longitud variable. Al final de la sentencia existe un checksum para la verificación e integridad de los datos. Este campo consta de un '\*' y dos dígitos hexadecimales que representan un OR exclusivo de 8 bits de todos los caracteres sin incluir el asterisco.

## 2.3 Interpretación de las sentencias

En esta sección se detalla cómo se logra extraer la información necesaria de las sentencias recibidas. Esta información nos indica una serie de campos que los transceptores AIS, instalados en las embarcaciones y diferentes estaciones costeras, transmiten y reciben usando tecnología TDMA.

Los receptores AIS ponen a disposición estos datos para sistemas anticolidión y otras tecnologías de ayuda a la navegación.

Dichos receptores procesan los paquetes de datos ASCII como un flujo de bytes utilizando conexiones serie como RS422-RS232 o USB y mediante Sockets a través de una red usando el formato NMEA 0183.

Los paquetes AIS tienen al inicio de la sentencia los caracteres ¡AIVDM los cuales pertenecen a los datos transmitidos por otros barcos y los caracteres ¡AIVDO que son las que transmiten el propio barco.

Mediante la recomendación de la Unión Internacional de Telecomunicaciones UIT, concretamente M.1371-5, las aclaraciones técnicas de IALA, y Organización Marítima Internacional podemos interpretar las características técnicas del sistema AIS.

En este apartado se verá solo lo que se encuentra en el alcance de este proyecto. Esta información será la interpretación de los campos necesarios para llevar a cabo nuestra aplicación.

Las sentencias están separadas mediante el delimitador “,” y cada campo tiene el siguiente significado:

- El primer campo representa la identificación de la sentencia. Como se ha mencionado nos encontramos con dos tipos principales:

AIVDM      Las transmitidas por otros buques.

AIVDO      Las transmitidas por el propio buque.

A continuación, se presenta un ejemplo de una sentencia AIVDM:

!AIVDM,1,1,,A,13e1@2001hOREmdDWSVcAI0j0H?P,0\*3A

- El segundo campo indica la cantidad de fragmentos que componen el mensaje. El protocolo NMEA 0183 limita a 82 caracteres el máximo que puede tener una sola sentencia con lo que, por ejemplo, para datos estáticos, necesitaremos más de un fragmento para poder interpretar toda la información.
- El tercer campo nos informa el número de fragmento que ocupa la sentencia en el mensaje completo.

- El cuarto campo representa una identificación para mensajes compuestos por más de una sentencia.
- El quinto campo indica un canal de radio de banda marina. Los valores pueden ser:
  - “A” que indica el canal 87B de frecuencia 161.975 de VHF.
  - “B” que indica el canal 88B de frecuencia 162.025 de VHF.
- El sexto campo contiene los datos dinámicos que utilizaremos para presentar la información.
- El séptimo campo nos indica el número de bits que se necesita para completar los datos en una codificación de 6 bits por carácter. Además, este campo contiene un (checksum) con el propósito de detectar cambios accidentales en la secuencia de datos y proteger la integridad de los mismos.

Para la obtención de los datos dinámicos nos centraremos en el campo 6. Los caracteres ASCII válidos utilizados en esta codificación son los siguientes:

Tabla 2

Símbolo	ASCII	Decimal	Bin
0	48	0	000000
1	49	1	000001
2	50	2	000010
3	51	3	000011
4	52	4	000100
5	53	5	000101
6	54	6	000110
7	55	7	000111
8	56	8	001000
9	57	9	001001
:	58	10	001010
;	59	11	001011
<	60	12	001100
=	61	13	001101
>	62	14	001110
?	63	15	001111
@	64	16	010000
A	65	17	010001
B	66	18	010010
C	67	19	010011
D	68	20	010100
E	69	21	010101
F	70	22	010110

Símbolo	ASCII	Decimal	Bin
G	71	23	010111
H	72	24	011000
I	73	25	011001
J	74	26	011010
K	75	27	011011
L	76	28	011100
M	77	29	011101
N	78	30	011110
O	79	31	011111
P	80	32	100000
Q	81	33	100001
R	82	34	100010
S	83	35	100011
T	84	36	100100
U	85	37	100101
V	86	38	100110
W	87	39	100111
`	96	40	101000
a	97	41	101001
b	98	42	101010
c	99	43	101011
d	100	44	101100
e	101	45	101101
f	102	46	101110
g	103	47	101111
h	104	48	110000
i	105	49	110001
j	106	50	110010
k	107	51	110011
l	108	52	110100
m	109	53	110101
n	110	54	110110
o	111	55	110111
p	112	56	111000
q	113	57	111001
r	114	58	111010
s	115	59	111011
t	116	60	111100
u	117	61	111101
v	118	62	111110
w	119	63	111111

Los datos se obtienen del campo 6 como agrupaciones de bits. Los tipos de datos que extraeremos sobre este campo son:

- Enteros sin signo.
- Enteros con signo.
- Strings.

Los datos se interpretan utilizando el formato Big Endian en el que los bits irán desde el más significativo al bit menos significativo y en complemento a2 donde, en los números positivos, el bit del extremo izquierdo es 0 y 1 para números negativos.

Los primeros seis bits representan el tipo de mensajes. A continuación, se describen los diferentes tipos según la recomendación ITU M.1371-5.

Tabla 3

Nº del mensaje	Nombre del mensaje	Comentario
0	No definido	
1	Informe de posición (programado)	
2	Informe de posición (asignado)	
3	Informe de posición (cuando se solicite)	
4	Informe de estación base	La Clase B «CS» debe obedecer la regla de 120 millas náuticas
5	Datos estáticos del barco y relacionados con la travesía	
6	Mensaje binario direccionado	
7	Acuse de recibo de mensajes binarios	
8	Mensaje binario de radiodifusión	
9	Informe normal de posición del avión de búsqueda y salvamento	
10	Petición de UTC y fecha	
11	Respuesta de UTC fecha	

<b>Nº del mensaje</b>	<b>Nombre del mensaje</b>	<b>Comentario</b>
12	Mensaje direccionado relacionado con la seguridad	NOTA – También se puede transferir Información haciendo uso del Mensaje 14
13		Debería transmitirse si se ha configurado la opción de procesar el Mensaje 12
14	Mensaje de radiodifusión relativo a la seguridad	
15	Interrogación	La Clase B «CS» debe dar respuesta a solicitudes del Mensaje 18 y del Mensaje 24
16	Instrucción de modo asignado	El Mensaje 23 es aplicable a la «CS»
17	Mensaje binario de radiodifusión del DGNSS	
18	Informe normal de posición de los equipos de la Clase B	El AIS Clase B «CS» debe indicar «1» para «detección de portadora» en el bit 143 de la bandera
19	Ya no se requiere informe ampliado de posición de los equipos de la Clase B	Transmítase ÚNICAMENTE como respuesta a una interrogación de la estación de base
20	Mensaje de gestión de enlace de datos	El Mensaje 4 debe recibirse y evaluarse para la regla de las 120 MN antes de responder
21	Informe de ayudas a la navegación	
22	Mensaje de gestión del canal	El uso de esta función puede ser diferente. Respuesta basada en las capacidades de la estación en ciertas regiones. No se aplica la regla de las 120 MN
23	Asignación de grupo	El Mensaje 4 debe recibirse y evaluarse para la regla de las 120 MN antes de responder
24	Datos estáticos de la Clase B «CS»	Parte A y Parte B



Nº del mensaje	Nombre del mensaje	Comentario
25	Mensaje binario de un solo intervalo	
26	Mensaje binario de varios intervalos con estado comunicaciones	
27	Informe de posición para aplicaciones de larga distancia	
28-63	Sin definir	Se reserva para uso futuro

En nuestro proyecto únicamente se controlará los mensajes de tipo 1 y tipo 2 para recoger los datos de posicionamiento (MMSI, latitud, longitud y rumbo) y los mensajes tipo 5 para obtener de los datos estáticos el nombre del buque.

#### **Sentencias de posición:**

Los mensajes de posición dinámicos son transmitidos periódicamente por todos los buques.

Los campos que utilizaremos para presentar la información son los siguientes:

Tabla 4

Parámetro	Número de bits	Campo	Descripción
ID de usuario	30	8,37	Número de MMSI
Longitud	28	68,88	Longitud en 1/10 000 de minuto (180°, Este = positivo, Oeste = negativo)
Latitud	27	89,115	Latitud en 1/10 000 de minuto (90°, Norte = positivo, Sur = negativo)
Rumbo	30	116,127	Rumbo

#### **Información estática:**

Los mensajes de tipo 5 (según tabla 2) nos proporcionarán información estática del buque de la cual extraeremos el nombre.

A continuación, mostramos la información estática transmitida:

Tabla 5

Parámetro	Número de bits	Campo	Descripción
Nombre del barco	120	112,231	20 caracteres de 6 bits ASCII como máximo

### 3. Análisis y diseño de la aplicación

En el siguiente apartado se presenta los requisitos, comportamiento y diseño que debe tener el sistema

#### 3.1 Definición de los requisitos:

La obtención de los requisitos se basa en la técnica de entrevistas y cuestionarios que se han realizado a unos imaginados stakeholders los cuales están interesados en una posible supervisión del tráfico marítimo.

Mediante una muestra representativa, los usuarios nos han transmitido una serie de necesidades y restricciones que el proyecto debe satisfacer. A continuación, se presenta los requisitos obtenidos:

#### Requisitos funcionales:

- El Receptor se podrá conectar a la red local mediante las siguientes opciones:
  - Utilizando una interfaz Ethernet cableada.
  - Utilizando una interfaz WIFI.
  - Mediante puerto serie y utilizando un conversor serial RS - 232 a Ethernet.
- La aplicación mostrará la información en tiempo real.
- La aplicación deberá distinguir entre las sentencias AIVDM/AIVDO y cualquier otra información recibida.
- Podrá representar información de hasta un máximo de 2000 barcos.
- La información mínima a representar será la posición del buque (latitud, longitud), rumbo y nombre de la embarcación.
- Los datos se deberán presentar mediante un mapa geográfico.
- El mapa se inicializará en la zona de cobertura del receptor.
- El mapa deberá permitir el movimiento a diferentes zonas posibilitando a su vez hacer zoom + y zoom – utilizando la rueda de desplazamiento del ratón.
- Los blancos permitirán distinguir, mediante un icono, el rumbo que un buque lleva en un determinado momento.

- Por cada barco, al hacer clic sobre el mismo, el sistema deberá presentar el MMSI, nombre y la posición en forma de popup.
- El sistema borrará un blanco cuando no se reciba información por más de diez minutos.

### Requisitos no funcionales:

- El sistema deberá procesar el flujo de datos de entrada a una velocidad de 38400 baudios.
- El sistema deberá leer las sentencias línea a línea sin retrasos indebidos. Cualquier retraso durante el procesamiento podría provocar la saturación del buffer de lectura del Socket cliente.
- La aplicación tendrá gran disponibilidad y será capaz de funcionar de forma ininterrumpida.
- El sistema deberá controlar estados incoherentes que puedan provenir del receptor.
- La aplicación debe cumplir con los requisitos de usabilidad marcados.

## 3.2 Diagrama de clases

Mediante el siguiente diagrama se presenta las clases que se utilizarán en el desarrollo de la aplicación.

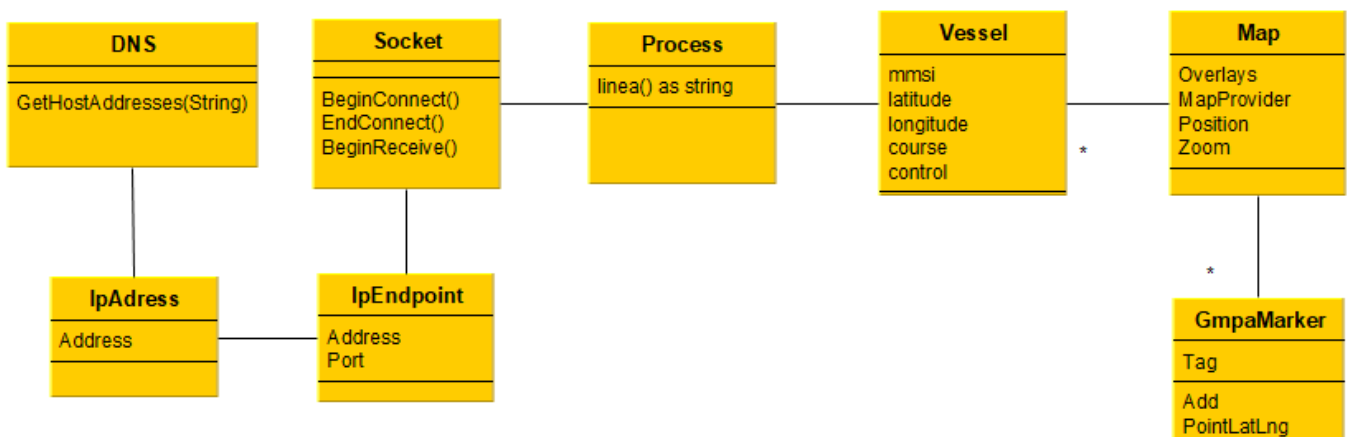


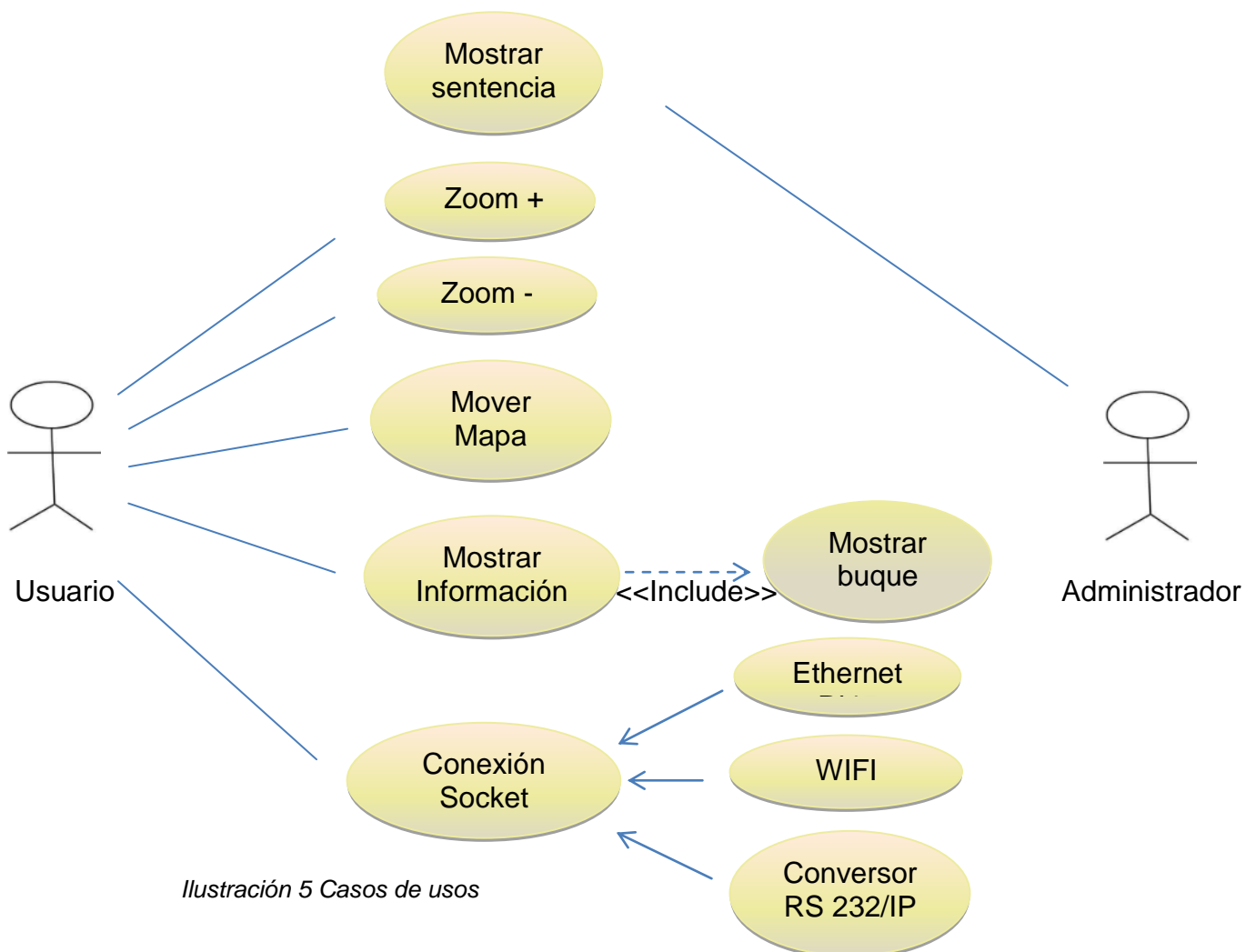
Ilustración 4 Diagrama de clases

Las clases DNS, Socket, IpAdress, IpEndpoint son clases utilizadas de la API .Net

- DNS se utiliza para resolver el nombre de dominio en caso de conectar desde internet.
- IpAdress e IpEndpoint proporcionan una dirección IP para pasarla posteriormente como parámetros a la clase DNS y Socket.
- Socket se usa para establecer una comunicación asíncrona con el receptor.
- La clase Map se utiliza para la representación gráfica de la información mediante un mapa geográfico.

### 3.3 Casos de uso

Mediante los casos de uso se describe la secuencia de acciones que los determinados actores pueden realizar.



## **Conexión:**

### Resumen:

El usuario se podrá conectar al receptor (servidor Socket) mediante un cliente Socket especificando la dirección IP y el puerto.

### Precondiciones:

El usuario deberá conocer la dirección IP y puerto del receptor AIS.

### Actores:

Usuario

### Descripción:

El usuario tendrá que especificar en una casilla la dirección IP correspondiente al receptor (servidor Socket).

También tendrá otra casilla en la que pondrá el puerto de destino.

Una vez rellenado estos campos podrá conectarse haciendo clic en el botón conectar.

## **Mover mapa:**

### Resumen:

El usuario podrá interactuar con el mapa geográfico.

### Actores:

Usuario

### Descripción:

El usuario puede mover el mapa para posicionarse utilizando el ratón haciendo clic en el mapa y desplazándolo.

## **Zoom +:**

### Resumen:

El usuario puede configurar manualmente el zoom + sobre el mapa. Esta opción puede resultar útil si queremos centrarnos en un determinado lugar de un mapa más grande.

### Actores:

Usuario

### Descripción:

El nivel de zoom sobre el mapa geográfico se configura automáticamente mostrando la zona de cobertura del receptor. No obstante, el usuario puede configurarlo manualmente mediante el scroll del ratón.

## **Zoom -:**

### Resumen:

De forma similar a la anterior, el usuario puede configurar el zoom -. Esta opción se utiliza para alejarse de una especificada zona y tener una visión más amplia del mapa geográfico.

### Actores:

Usuario

### Descripción:

El usuario puede configurar manualmente mediante el scroll del ratón moviéndolo hacia adelante.

## **Mostrar información:**

### Resumen:

El usuario podrá ver representado en el mapa geográfico la información procedente de los buques.

### Precondiciones:

Deberemos estar conectados al receptor mediante la conexión Socket.

Actores:

Usuario

Descripción:

La información presentada en el mapa geográfico es la posición del buque y el rumbo que está llevando en tiempo real.

Esta información es interpretada mediante un icono identificativo el cual hace posible identificar el rumbo que el barco está tomando.

**Mostrar datos buque:**

Resumen:

El usuario puede ver la información procedente de cada buque.

Precondiciones:

Deberemos estar conectados al receptor mediante la conexión Socket.

Incluye:

Para mostrar la información detallada necesitamos el caso “mostrar información”.

Actores:

Usuario

Descripción:

Haciendo clic sobre cada barco, el usuario puede ver mediante un mensaje popup la información procedente de cada buque.

Esta información es:

- Su MMSI (Maritime Mobile Service Identity). Número de identificación del barco.
- Posición (Latitud, Longitud)
- El nombre del buque.

## Mostrar sentencias

### Resumen:

En modo administrador tenemos la posibilidad de ver los datos que se está recibiendo desde el receptor.

### Precondiciones:

Deberemos estar conectados al receptor mediante la conexión Socket.

Es necesario conocer la contraseña de administrador.

### Actores:

Administrador

Descripción:

Mediante la barra de menús podemos entrar en modo depuración mediante el cual podemos visualizar las sentencias que van llegando desde el receptor. Para poder entrar seleccionaremos la opción configuración-depurar.

Una vez aparezca la ventana introducimos la contraseña de administrador: "12345".

Finalmente aparecerá una ventana en el formulario principal donde veremos las sentencias que van llegando.

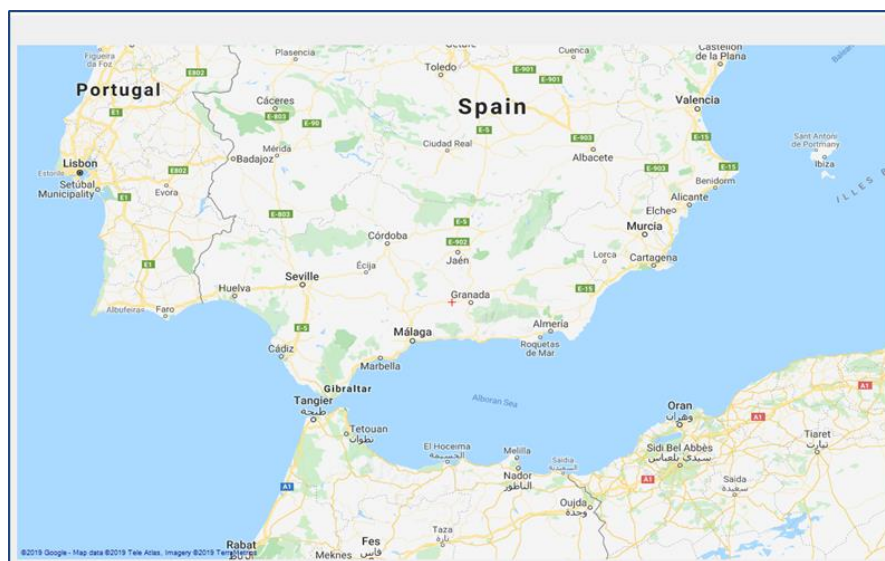


## 4. Implementación

La fase de implementación se ha dividido en una pequeña introducción donde se describe el diseño a implementar, el comportamiento del sistema y cuatro principales bloques los cuales han sido fundamentales en el desarrollo de la aplicación. El orden y la ejecución de cada uno de estos han sido determinantes para la consecución de los objetivos marcados.

### Definición del diseño a implementar:

- El receptor AIS que se utilizará será el modelo QK-A027 de la marca QUARK-ELEC de doble canal para las frecuencias VHF.
- El sistema se implementará mediante una aplicación de escritorio utilizando .NET framework.
- El IDE que se utilizará para el desarrollo de la aplicación será Visual Studio 2019.
- Como mapa geográfico utilizaremos el componente libre y multiplataforma GMap.NET el cual nos proporcionará geolocalización en tiempo real.
- La ventana principal de la aplicación mostrará el mapa geo-referenciado. Además, dicha ventana contiene información relativa a la conexión Socket como la dirección IP del servidor, puerto y los botones de conexión y desconexión.



*Ilustración 6 Mapa geográfico*

- Se crearán iconos personalizado de manera que nos permita saber de una manera rápida el rumbo que sigue un buque. Para

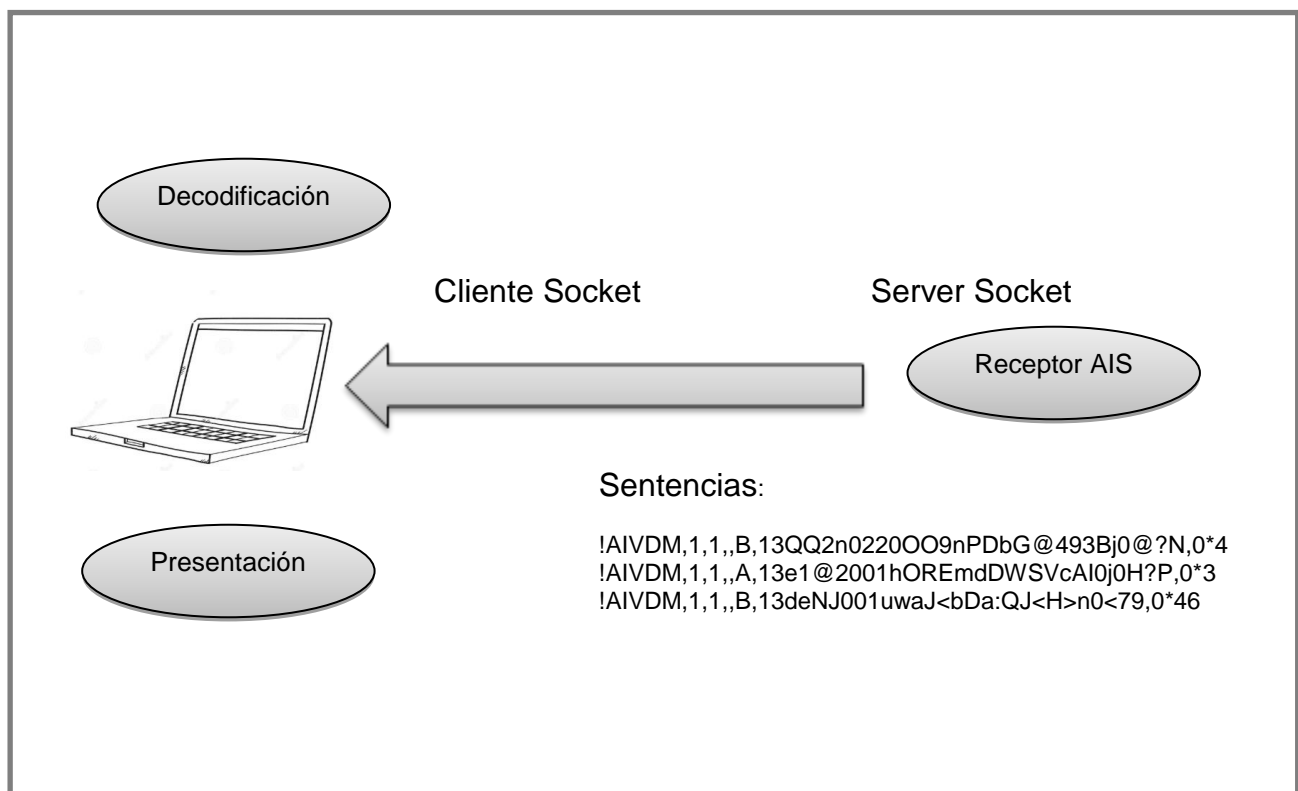
lograr este fin utilizaremos la herramienta de creación y edición de imágenes GIMP 2.

### Comportamiento del sistema:

El receptor obtendrá las transmisiones en tiempo real de las embarcaciones que utilizan un transpondedor AIS utilizando canales de banda marina. Posteriormente, y en nuestro caso, el receptor envía las sentencias usando el protocolo NMEA0183, mediante un servidor Socket, a la aplicación.

Esta última, deberá decodificar dicha información y estructurarla de manera que posteriormente se pueda representar en un mapa geográfico.

Finalmente, el usuario podrá monitorizar un escenario de tráfico marítimo en tiempo real.



*Ilustración 7 Comportamiento del sistema*

A continuación, se detalla cada uno de los bloques que componen la etapa de implementación:

#### 4.1 Conexión con el receptor

Para la conexión con el receptor AIS, la aplicación ha utilizado un Socket como origen de la comunicación (cliente Socket). En el caso del receptor como destino utiliza un servidor Socket. Para establecer este Socket lo único que necesitamos saber es la dirección IP del servidor Socket y en este caso, también asignaremos el número de puerto.

El estilo de la comunicación es una secuencia de bytes en el que los datos se van transmitiendo como un flujo de bytes de extremo a extremo de la comunicación. En este estilo, el protocolo de transporte garantiza el envío de los datos en el mismo orden en el que se han transmitido sin pérdidas y sin repeticiones.

El protocolo utilizado en este estilo de comunicación es el TCP, por tanto, puesto que la comunicación consiste en establecer previamente la conexión con el Socket remoto, tenemos una comunicación orientada a conexión. Una vez realizada dicha conexión comenzaremos a recibir datos en un flujo ordenado.

La comunicación entre ambas partes realiza las siguientes operaciones:

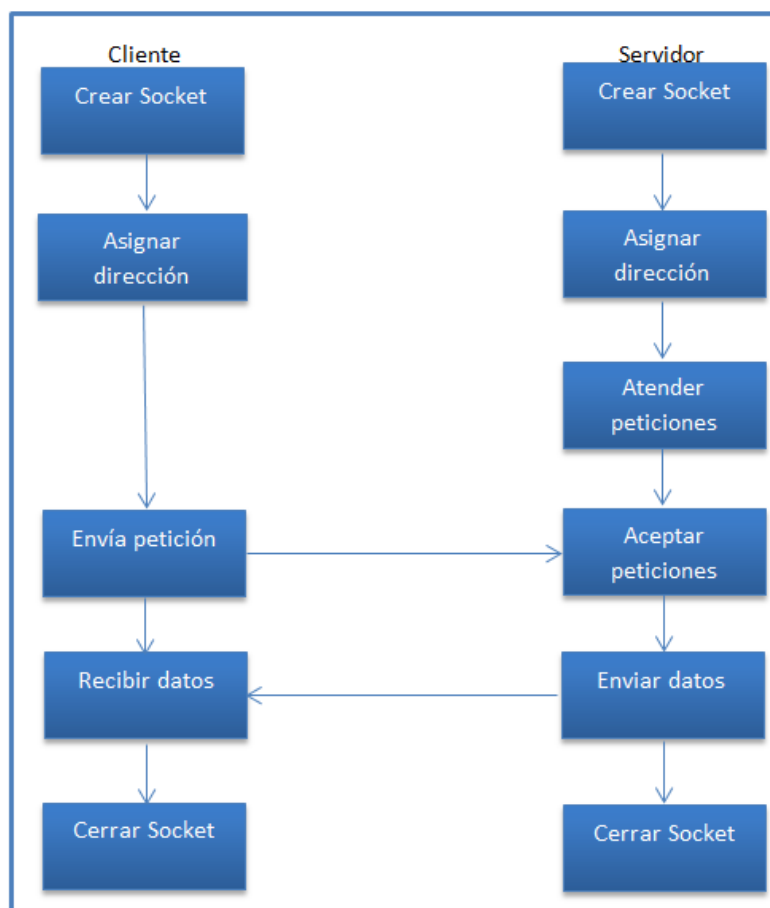


Ilustración 8 Cuadro resumen de las operaciones

Para implementar estas operaciones, se utiliza la clase Socket de .NET Framework. A continuación, se detalla cómo se programan dichas operaciones en la aplicación:

Para crear el socket se especifica el esquema de direccionamiento y el protocolo deseado. En este caso TCP

```
clientSocket = New Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp)
Dim ipAddress As IPAddress = IPAddress.Parse(myIPs(0).ToString)
Dim ipEndPoint As IPEndPoint = New IPEndPoint(ipAddress, CInt(Portin.Text))
clientSocket.BeginConnect(ipEndPoint, New AsyncCallback(AddressOf ConnectSk), Nothing)
buttoncontrol = True
Ipin.Enabled = False
Portin.Enabled = False
```

*Ilustración 9 Crear cliente socket*

Posteriormente, se asigna la dirección IP y número de puerto.

```
clientSocket = New Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp)
Dim ipAddress As IPAddress = IPAddress.Parse(myIPs(0).ToString)
Dim ipEndPoint As IPEndPoint = New IPEndPoint(ipAddress, CInt(Portin.Text))
clientSocket.BeginConnect(ipEndPoint, New AsyncCallback(AddressOf ConnectSk), Nothing)
buttoncontrol = True
Ipin.Enabled = False
Portin.Enabled = False
```

*Ilustración 10 Dirección IP y Puerto*

Mediante el método BeginConnect enviamos una solicitud asíncrona para la conexión con el receptor.

```
clientSocket = New Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp)
Dim ipAddress As IPAddress = IPAddress.Parse(myIPs(0).ToString)
Dim ipEndPoint As IPEndPoint = New IPEndPoint(ipAddress, CInt(Portin.Text))
clientSocket.BeginConnect(ipEndPoint, New AsyncCallback(AddressOf ConnectSk), Nothing)
buttoncontrol = True
Ipin.Enabled = False
Portin.Enabled = False
```

*Ilustración 11 Envía petición*

Una vez aceptada la solicitud el receptor comienza a enviar datos. Estos datos se recogen mediante un array de tipo byte.

```
Labelconnectdis.Text = "Conectado"
Dim bytesRec As Byte() = dataByte
message &= System.Text.ASCIIEncoding.ASCII.GetString(dataByte)
clientSocket.BeginReceive(dataByte, 0, dataByte.Length, SocketFlags.None,
    New AsyncCallback(AddressOf Recieve), clientSocket)
```

*Ilustración 12 Recibir datos*

Por último, cuando se quiera cerrar la conexión socket se llama al método close.

```
If clientSocket.Connected = True Then
    Try
        clientSocket.Disconnect(True)
        clientSocket.Close()
```

*Ilustración 13 Cerrar socket*

La mayor dificultad en esta fase la supuso controlar el flujo de información procedente del receptor. En un principio, tal y como estaba diseñado, saturaba el buffer de lectura y se cortaba la comunicación. Finalmente, se pudo solucionar declarando un array de tipo byte con una sola posición "dataByte". Una vez obtenido cada byte, lo iba concatenando usando una variable de tipo String "message".

## 4.2 Procesamiento de la información

Una vez establecida la conexión, el receptor estará continuamente enviando sentencias NMEA a la aplicación. En esta fase se procesa la información con la finalidad de ir obteniendo línea a línea, y una vez obtenida, la dividiremos en campos.

Los datos van llegando y se van almacenando en una variable de tipo byte.

Esta información se irá codificando desde tipo byte a cadena en formato ASCII.

La variable "message" guarda cada carácter codificado y posteriormente llama al procedimiento line.

```
message &= System.Text.ASCIIEncoding.ASCII.GetString(dataByte)

clientSocket.BeginReceive(dataByte, 0, dataByte.Length, SocketFlags.None,
    New AsyncCallback(AddressOf Recieve), clientSocket)

Line()
```

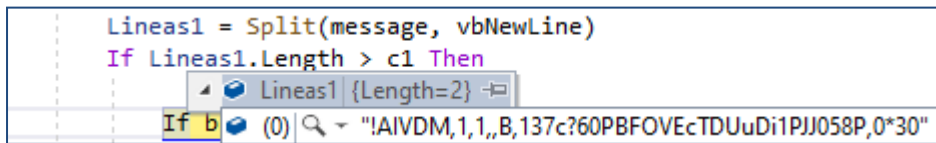
*Ilustración 14 Codificación array de bytes a string*

En el procedimiento Line(), mediante el método Split, podemos dividir los mensajes cuando la secuencia llegue a una nueva línea mediante el limitador vbNewLine.

```
process.line = Split(message, vbNewLine)
If process.line.Length > c1 Then
```

*Ilustración 15 Separación cadena de texto por línea*

Por último, en caso de que el array de líneas tenga más de un elemento significará que ya tenemos la línea y pasaremos a la fase siguiente donde se interpreta la sentencia.



```
Lineas1 = Split(message, vbNewLine)
If Lineas1.Length > 1 Then
    Lineas1 {Length=2} =>
    If b (0) (0) "IAIVDM,1,1,,B,137c?60PBFOVEcTDUuDi1PJJ058P,0*30"
```

Ilustración 16 Array mayor de un elemento y por tanto obtención de la línea

### Definición de las estructuras de datos a utilizar:

Usaremos un tipo de datos como la clase para almacenar los distintos valores relacionados para cada barco y las distintas líneas que se van procesando.

Esta estructura nos permitirá asociar las diferentes propiedades a un objeto. También, nos da la posibilidad de utilizar colecciones dinámicamente.

A continuación, se presenta las clases definidas:

Vessel {	Process {
Mmsi:	line()
Latitude:	}
Longitude:	
Name:	
Course:	
Control	
}	

Además, necesitamos una estructura que nos permita insertar y borrar las operaciones según si un barco entra en cobertura o sale fuera de ésta. Por lo tanto, se utilizará una colección dinámica como `ArrayList` para almacenar los buques, marcadores y las capas del mapa. En nuestro caso corresponden las siguientes colecciones:

`overlayOne`. Representa las diferentes capas del mapa donde se crean marcadores

`Lvessel`. Contiene los datos de los buques

`Lmarker`. Contiene los marcadores que representan a cada buque.

### 4.3 Decodificación de la información

Tal y como se ha comentado anteriormente, para los datos AIS, el estándar NMEA utiliza dos principales tipos de sentencias:

- AIVDM para datos recibidos de otros barcos.
- AIVDO para datos procedente del propio barco.

En nuestro caso, para representar la información de los demás buques, se hará una distinción mediante una estructura de control cada vez que se reciba una sentencia con la finalidad de identificar solo los mensajes AIVDM. También, se hará distinción entre los mensajes de tipo 1 para datos dinámicos y tipo 5 para datos estáticos.

Una vez divida la sentencia en campos será el campo 6 el que nos proporcione la información dinámica y estática.

Lo primero de todo será pasar este campo a un array de bits. Para ello se crea una variable de tipo BitArray inicializado a “false” que contendrá, una vez pasado a binario cada carácter, un array con valor true donde hay un uno y false donde hay un cero.

```
'Pasamos de String a bit.  
2 referencias  
Private Function Stringbit(ByVal cad As String) As BitArray  
    Dim chr As Char() = cad.ToCharArray  
    Dim bita As New BitArray(chr.Length * 6, False)  
    Dim a0 As Integer = 0
```

Ilustración 17 Función que devuelve un BitArray

Ya que cada carácter se codifica con 6 bits, multiplicaremos el tamaño del array por 6.

(0)	False	(15)	True	(30)	False
(1)	False	(16)	False	(31)	False
(2)	False	(17)	True	(32)	False
(3)	False	(18)	False	(33)	False
(4)	False	(19)	True	(34)	True
(5)	True	(20)	False	(35)	False
(6)	False	(21)	False	(36)	False
(7)	False	(22)	False	(37)	True
(8)	False	(23)	False	(38)	False
(9)	False	(24)	True	(39)	False
(10)	True	(25)	True	(40)	False
(11)	True	(26)	True	(41)	False
(12)	False	(27)	False	(42)	False
(13)	True	(28)	True	(43)	False
(14)	True	(29)	True	(44)	False

Ilustración 18 Bitarray



Posteriormente, en función del campo que se quiera obtener, tenemos 3 funciones las cuales realizan las siguientes operaciones:

- Pasar de bit a entero
- Pasar de bit a entero con signo
- Pasar de bit a ASCII

Para campos de tipo entero positivos como por ejemplo MMSI y course pasaremos de tipo bit a int.

Para ello, mediante una operación and con desplazamiento de bits a la izquierda y usando una máscara, sabremos los bits que están a uno.

```
000001000010001100011000011000110000110001001010010010010
<< 1
000001000010001100011000011000110000110001001010010010010
<< 1
000001000010001100011000011000110000110001001010010010010
<< 1
```

Finalmente pasaremos el resultado a un valor de tipo entero.

```
Dim recorre As Integer = 0
For I As Integer = desde To desde + n - 1
    recorre = (recorre << 1) Or IIf(a(I), 1, 0)
Next I
Return recorre
```

*Ilustración 19 Desplazamiento de bit a la izquierda*

Para campos que representen número enteros negativos como por ejemplo la posición, se necesitan distinguir cuando el bit más significativo es uno (para números negativos) y cero para positivos.

```
If Not abit(desde - 1) Then
    'Positivo
    desde -= 1
    Dim recorre As Integer = 0
    For I As Integer = desde To desde + n - 1
        recorre = (recorre << 1) Or IIf(abit(I), 1, 0)
    Next I
    Return recorre
Else
    'Negativo
    Dim recorre As Integer = 0
    For I As Integer = desde To desde + n - 2
        recorre = (recorre << 1) Or IIf(abit(I), 0, 1)
    Next I
    Return -recorre - 1
```

*Ilustración 20 Pasa de bit a entero con signo.*



Por último, para valores que representen caracteres ASCII, como por ejemplo el nombre del buque se pasará de bit a String. Para lograrlo, se realiza un tratamiento similar (desplazando los bits para saber los que están a uno) y posteriormente pasaremos a cadena el código ASCII.

```

For k As Integer = i To i + 5
    t = (t << 1) Or IIf(abit(k), 1, 0)
    cuentabucle += 1
Next k
If t < 32 Then t += 64
TmpStr.Append(Chr(t))
Next

```

*Ilustración 21 Pasa de bit a ASCII*

A continuación, se presenta un ejemplo como resultado de aplicar los procedimientos descritos:

#### **Datos dinámicos:**

!AIVDM,1,1,,A,13Vr@w002`OW@@`DWAePgPS600Rp,0\*0A

MMSI: 242127100  
 Latitude: 36.025050°  
 Longitude: -5.405832°  
 Rumbo : 17°

!AIVDO,1,1,,B,4028ipQv>O;@PwVLddDW0aG00D5L,0\*18

MMSI: 2240994  
 Latitud: 36.017768  
 Longitud: -5.581830

#### **Datos estáticos:**

!AIVDM,2,1,9,B,53KA`F82D9AWTP7?O;@4j0@4Pp40000000000017Ckjc`6Qe0VQhTkm@,0\*7A  
 !AIVDM,2,2,9,B,0000000000000000,2\*2E

MMSI: 229927000  
 Nombre : AL DAHNA

*Ilustración 22 Ejemplo de sentencias AIVDM, AIVDO y su decodificación:*

Esta fase fue la que más tiempo me ocupó y la que supuso mayores problemas. No tenía mucha experiencia en programación realizando operaciones con bits. Lo que más me ayudó fue el tener claro sobre el diseño de lo que había que hacer. Antes de pasar a la implementación realicé muchas pruebas prácticas sobre el papel.

## 4.4 Representación de la información

### Mapa geográfico

Para la representación de la información usaremos el componente gratuito GMAP.NET el cual instalaremos en nuestro proyecto.

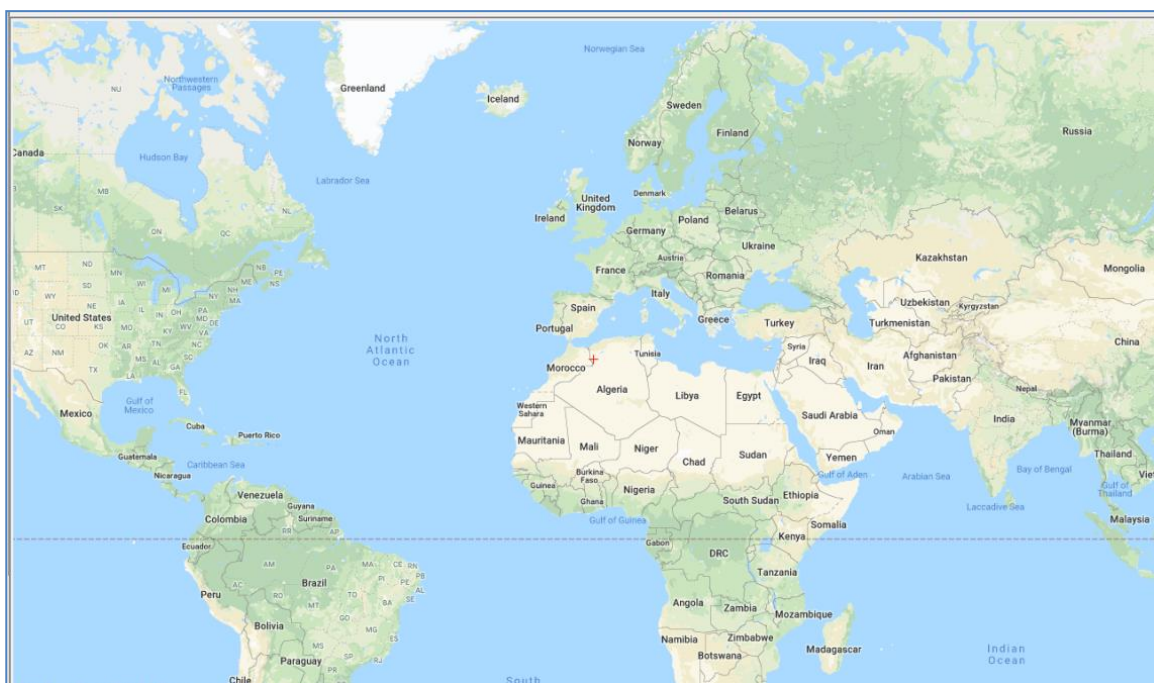
Este componente permite la captura de datos en tiempo real. Además, permite la actualización dinámica de las capas de información mediante una interfaz de usuario intuitiva y fácil de usar.

La información se estructura en capas y se encuentra geo-posicionada.

El componente instalado en nuestro proyecto es:

#### ➤ GMap.Net.Windows

Como proveedor de datos se ha utilizado Google Maps. Las coordenadas se encuentran en el sistema WGS84 en latitud y longitud, positiva para los cuadrantes Norte y Este y negativa para Sur y Oeste.



*Ilustración 23 Gmap.net*

El mapa utiliza capas “overlays” y es en ésta donde se insertan los marcadores.

## Inicialización mapa

Una vez insertado el componente en el formulario, pasamos a la inicialización del mapa mediante la definición de las siguientes propiedades:

```
Private Sub GMapControl1_Load(sender As Object, e As EventArgs) Handles GMapMain.Load
    GMapMain.DragButton = MouseButtons.Left
    GMapMain.CanDragMap = True
    GMapMain.MapProvider = GMapProviders.GoogleMap
    GMapMain.Position = New PointLatLng(36.031656, -5.299825)
    GMapMain.MinZoom = 0
    GMapMain.MaxZoom = 24
    GMapMain.Zoom = 11
    GMapMain.AutoScroll = True
    GMapMain.Overlays.Add(overlayOne)
End Sub
```

*Ilustración 24 Parámetros de inicialización del mapa.*

## Creación, actualización y borrado de marcadores

Con los campos ya decodificados se pasa a crear los marcadores en el mapa. Para ello se necesita saber, por cada sentencia que llega, si el buque se encuentra en el arraylist o si se crea por primera vez.

```
Private Sub AddMarker()
    Dim marker As GMapMarker
    If found = True Then
        Try
            overlayOne.Markers(i2) = New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New PointLatLng(Lvessel(i2).latitude, Lvessel(i2).longitude), imgship)
            overlayOne.Markers(i2).Tag = (i2)
        Catch ex As Exception
        End Try
    Else
        marker = (New GMap.NET.WindowsForms.Markers.GMarkerGoogle(New PointLatLng(Lvessel(i1).latitude, Lvessel(i1).longitude), imgship))
        marker.Tag = (i1)
        overlayOne.Markers.Add(marker)
        Lmarker.Add(marker)
    End If
End Sub
```

*Ilustración 25 Procedimiento que crea o actualiza los marcadores*

Para el borrado de marcadores se utiliza una variable de control dentro de la clase Vessel y un timer que controla periodos de 10 minutos.

La variable booleana control se inicializa con el valor a false. Transcurrido los primeros 10 minutos se pasa el valor a true para todos los buques. Posteriormente, conforme se reciben nuevas posiciones se pasa de nuevo el valor a false con lo que, pasados los siguientes 10 minutos, los que quedaron a true son los buques de los que no se han recibido nuevas posiciones y por tanto, hay que borrar.

```

If Larrayaborrar.Count > 0 Then
    For ind2 = Larrayaborrar.Count - 1 To 0 Step -1
        overlayOne.Markers.RemoveAt(Larrayaborrar(ind2))
        Lvessel.RemoveAt(Larrayaborrar(ind2))
        Lmarker.RemoveAt(Larrayaborrar(ind2))
    Next
End If
Larrayaborrar.Clear()

```

*Ilustración 26 Borrado de buques*

Uno de los problemas en la representación gráfica, lo supuso el rotar los marcadores en función de un ángulo que, en nuestro caso, es el rumbo que recibimos del buque. Para implementar correctamente este requerimiento se creó una función la cual utiliza las propiedades del objeto Bitmap con el fin de conseguir nuestro objetivo.

```

Public Function Rotate(img As Image, course As Single) As Bitmap
    Dim rotatedImg As New Bitmap(img.Width, img.Height)
    rotatedImg.SetResolution(img.HorizontalResolution, img.VerticalResolution)
    Dim icon As Graphics = Graphics.FromImage(rotatedImg)
    Dim coord As PointF = New PointF(img.Width / 2, img.Height / 2)
    icon.TranslateTransform(coord.X, coord.Y)
    icon.RotateTransform(course)
    icon.TranslateTransform(-coord.X, -coord.Y)
    icon.DrawImage(img, New PointF(0, 0))
    Return rotatedImg
End Function

```

*Ilustración 27 Función para rotar una imagen en función del rumbo.*

## 5. Pruebas

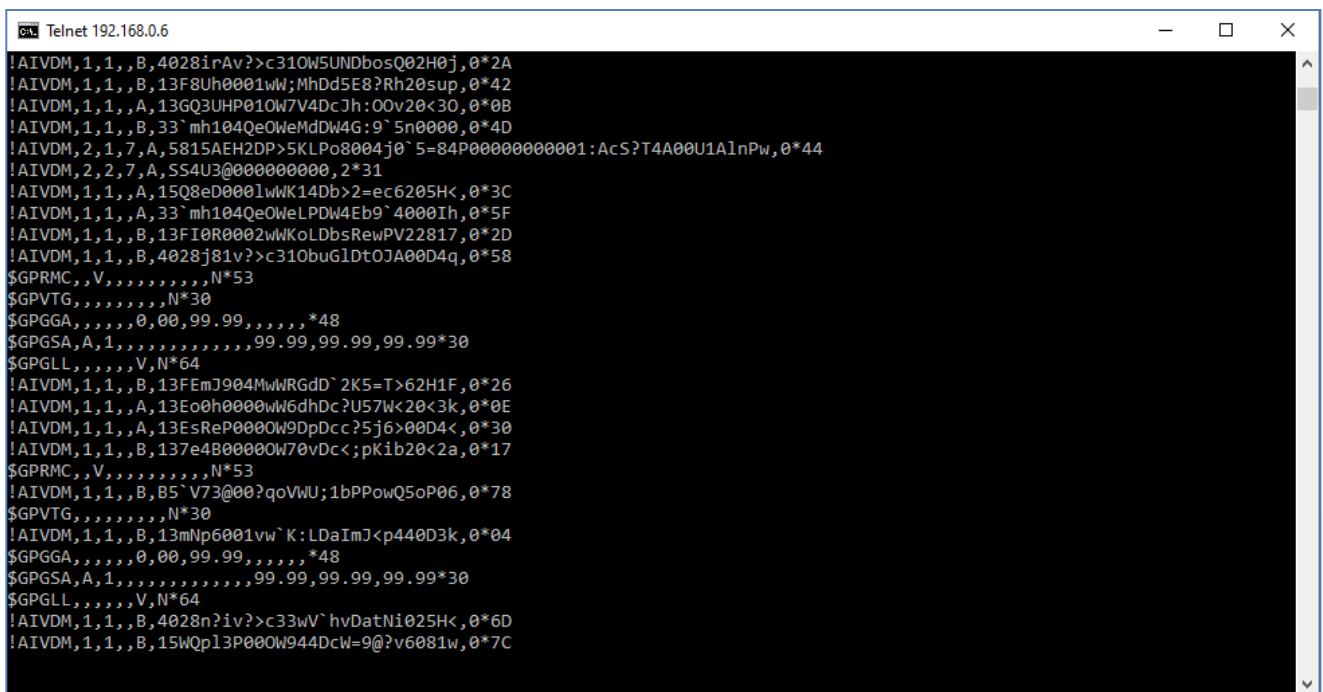
Esta fase se dividirá en unas series de pruebas. Por un lado, se realizarán las pruebas referentes a la conexión, comunicación y procesamiento de la información y por otro lado se comprobará la parte operativa relacionada con el funcionamiento de la aplicación.

### Envío de sentencias

- Se comprueba que el receptor envía las sentencias cuando se le solicita. Esta prueba es ajena a la aplicación y significa una buena herramienta para la detección de fallos.

Prueba: Telnet 192.168.0.6 2000

Respuesta: Visualización de las sentencias



```
Telnet 192.168.0.6
!AIVDM,1,1,,B,4028irAv?>c310W5UNDbosQ02H0j,0*2A
!AIVDM,1,1,,B,13F8Uh0001wW;MhDd5E8?Rh20sup,0*42
!AIVDM,1,1,,A,13GQ3UHP010W7V4DcJh:00v20<30,0*0B
!AIVDM,1,1,,B,33`mh104QeOWeMdW4G:9`5n0000,0*4D
!AIVDM,2,1,7,A,5815AEH2DP>5KLPo8004j0`5=84P0000000001:AcS?T4A00U1AInPw,0*44
!AIVDM,2,2,7,A,SS4U3@000000000,2*31
!AIVDM,1,1,,A,15Q8eD0001wWK14Db>2=ec6205H<,0*3C
!AIVDM,1,1,,A,33`mh104QeOWeLPDW4Eb9`4000Ih,0*5F
!AIVDM,1,1,,B,13FI0R0002wWkoLDbsRwPV22817,0*2D
!AIVDM,1,1,,B,4028j81v?>c310buG1Dt0JA00D4q,0*58
$GPRMC,V,,,,,,,,,N*53
$GPVTG,,,,,,,,,N*30
$GPGGA,,,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,,V,N*64
!AIVDM,1,1,,B,13FEmJ904MwWRGdD`2K5=T>62H1F,0*26
!AIVDM,1,1,,A,13Eo0h0000wW6dhDc?U57W<20<3k,0*0E
!AIVDM,1,1,,A,13EsReP0000W9DpDcc?5j6>00D4<,0*30
!AIVDM,1,1,,B,137e4B0000W70vDc<;pKib20<2a,0*17
$GPRMC,V,,,,,,,,,N*53
!AIVDM,1,1,,B,B5`V73@00?qoVWU;1bPPowQ5oP06,0*78
$GPVTG,,,,,,,,,N*30
!AIVDM,1,1,,B,13mNp6001vw`K:LDaImJ<p440D3k,0*04
$GPGGA,,,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,,V,N*64
!AIVDM,1,1,,B,4028n?iv?>c33wV`hvDatNi025H<,0*6D
!AIVDM,1,1,,B,15WQp13P000W944DcW=9@?v6081w,0*7C
```

Ilustración 28 Visualización sentencias usando Telnet

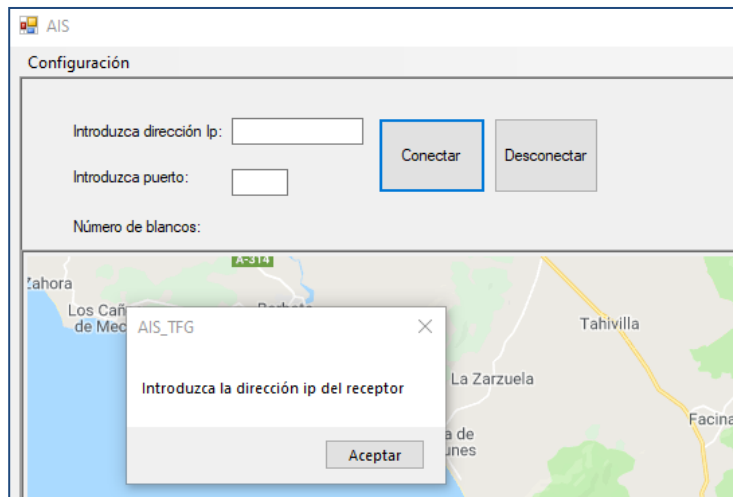
Resultado: Correcto

## Conexión

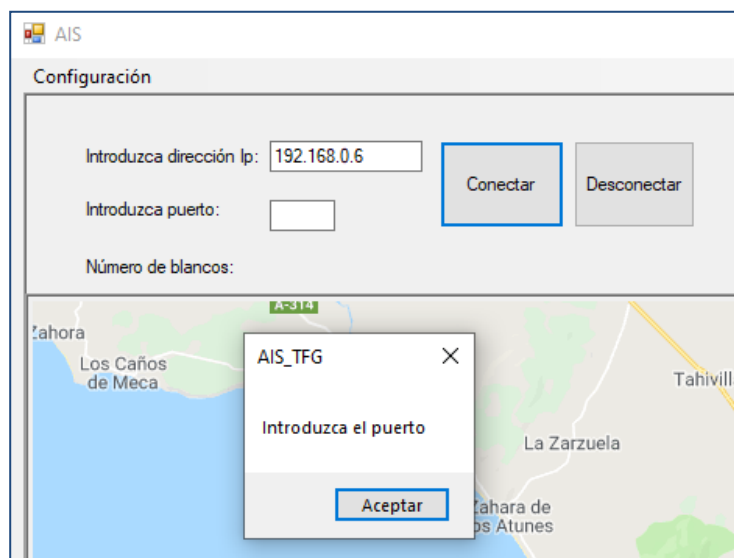
- En este apartado se probará la conexión con el receptor.

Prueba: Se introduce una IP o puerto no válido

Respuesta: En caso de no especificar ninguno de los datos o introducir alguno erróneo la aplicación contestará especificando el error.



*Ilustración 29 Si no introducimos IP*



*Ilustración 30 Si no introducimos puerto*

Resultado: Correcto

## Comunicación

- Se comprueba la comunicación con el receptor

Prueba: Se activa modo debug en el menú configuración.

Respuesta: Visualización de las sentencias

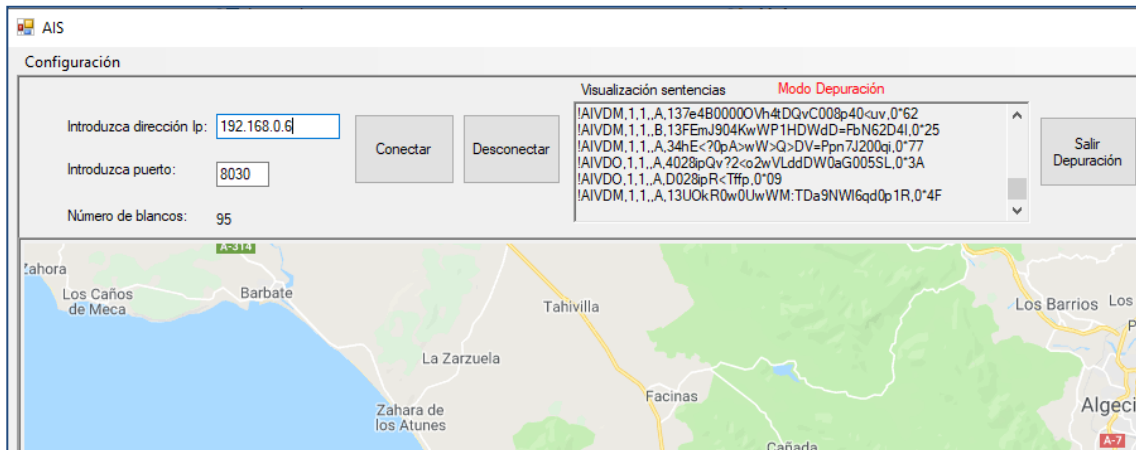


Ilustración 31 Visualización de las sentencias

Resultado: Correcto

## Procesamiento

- Comprobación de la codificación de las sentencias. Se observa en la lista de buques con los datos decodificados

Prueba: Se activa modo debug en el menú configuración.

Respuesta: Pasados 10 minutos se visualizan las sentencias decodificadas actualizadas.

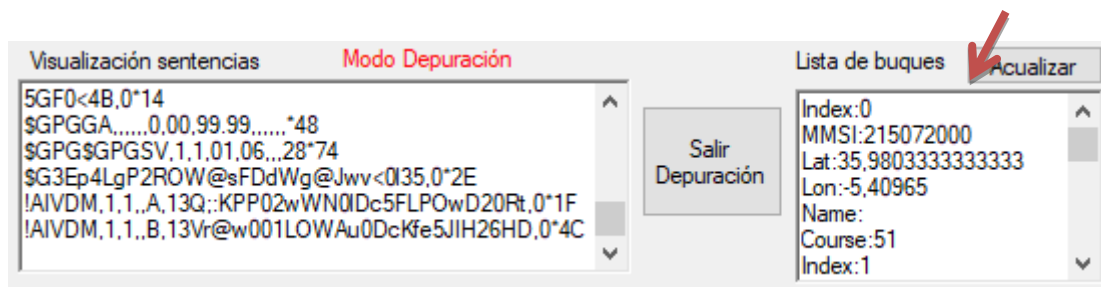


Ilustración 32 Lista con datos procesados

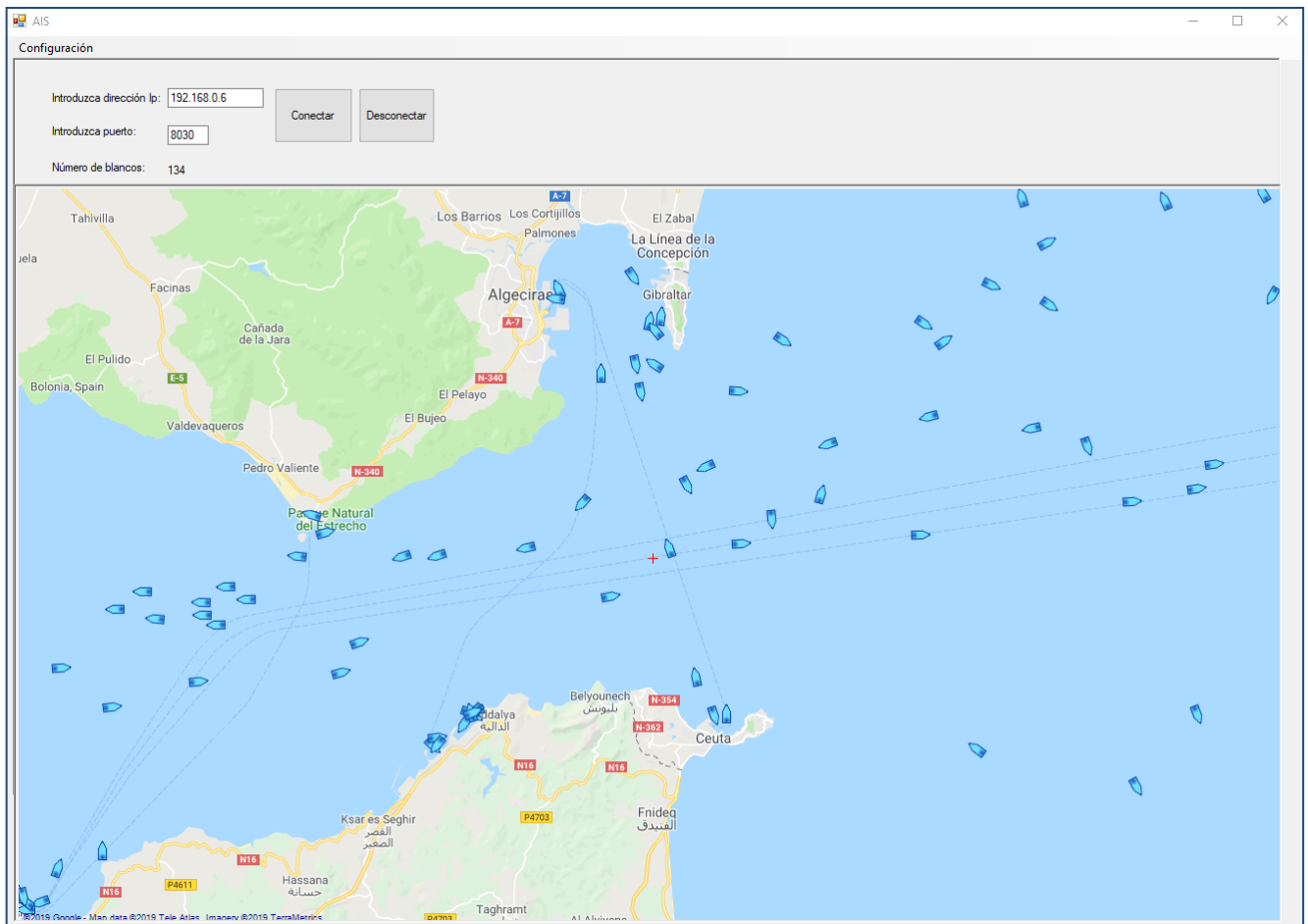
Resultado: Correcto

## Operativa

- Consistencia. Comprobación del comportamiento de la aplicación

**Prueba:** Se deja la aplicación funcionando durante un periodo largo de tiempo y se comprueba que trabaja correctamente sin comportamientos indeseados y sin retrasos en la recepción de los datos.

**Respuesta:** Representación del tráfico marítimo




*Ilustración 33 Aplicación en operación*

**Resultado:** Correcto



## 6. Manual de usuario

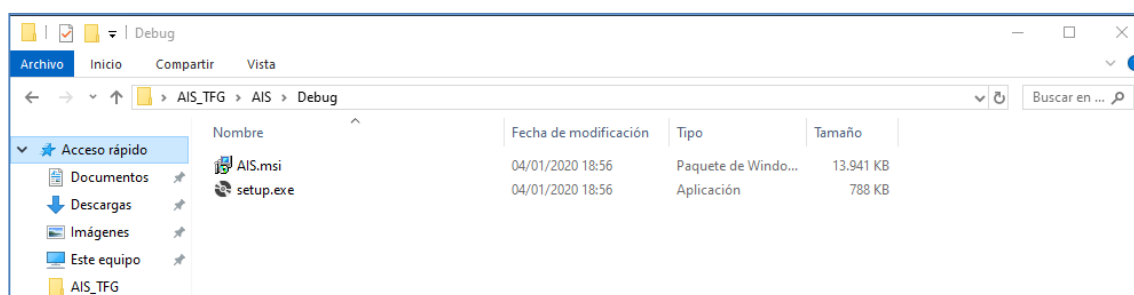
Lo primero de todo será descomprimir el fichero AIS\_TFG:

Nombre	Fecha de modificación	Tipo	Tamaño
 AIS_TFG.zip	02/01/2020 12:19	Archivo WinRAR Z...	68.898 KB

*Ilustración 34 Archivo a descomprimir*

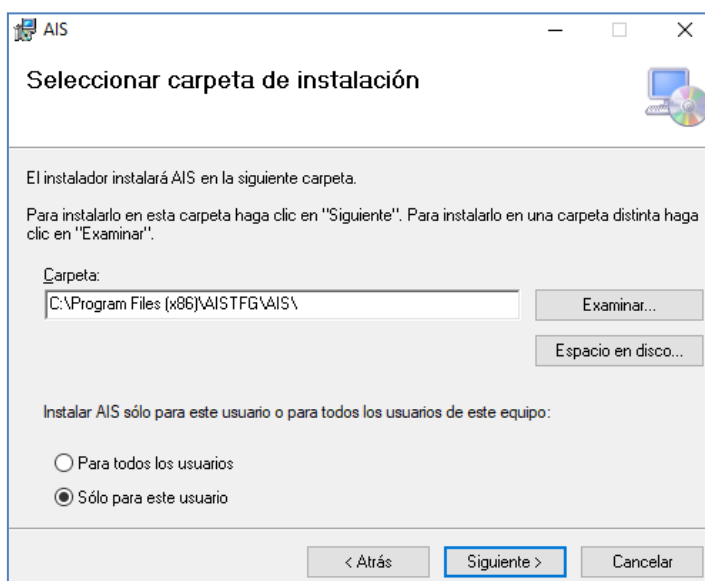
### Instalación

Una vez descomprimido, navegaremos hasta la carpeta AIS\_TFG\AIS\Debug, y una vez allí ejecutaremos el archivo setup.exe para dar comienzo al proceso de instalación.



*Ilustración 35 Archivo setup.exe*

Seguidamente se elige la carpeta donde se quiere instalar y los usuarios para los que se quiere instalar:



*Ilustración 36 Carpeta de instalación*

Por último, se confirma el proceso de instalación.

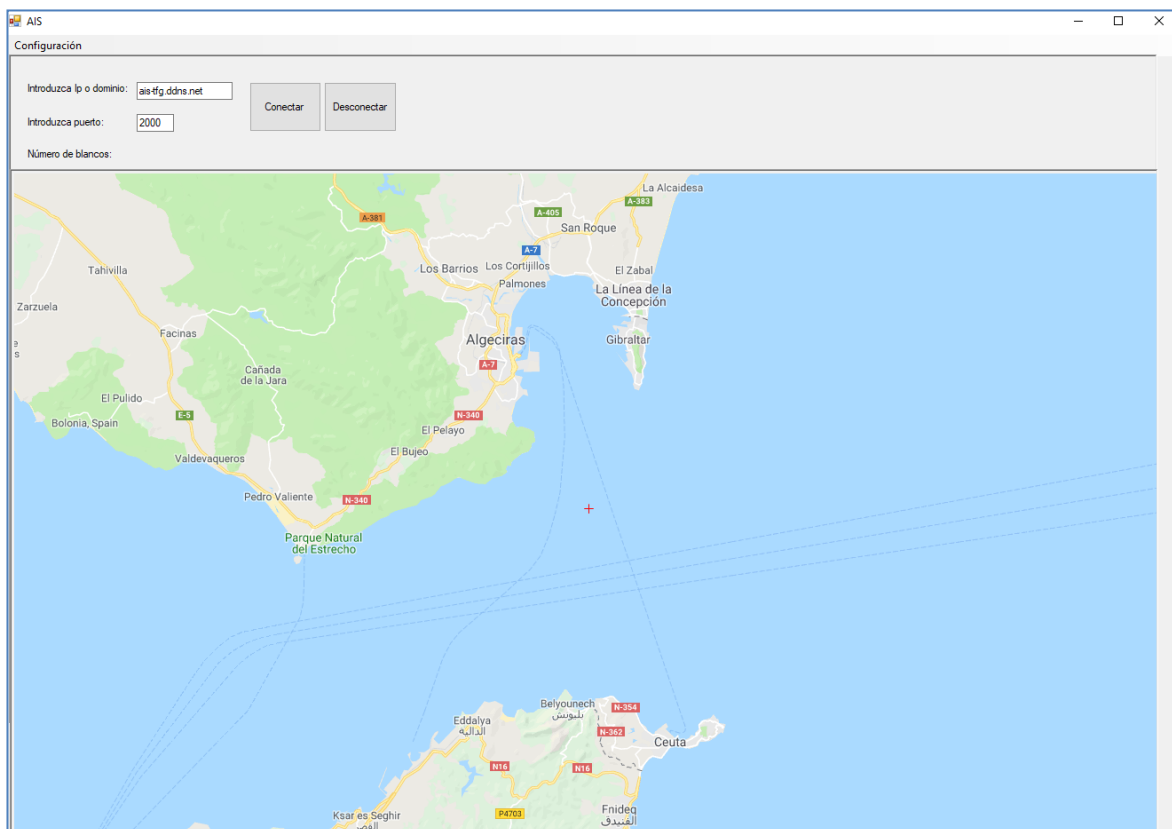
## Uso

Es importante, con el fin de cargar las capas del mapa y resolver nombres de domino, ejecutar la aplicación desde un ordenador con conexión a internet.

Una vez instalado, se ejecuta la aplicación desde el menú de inicio o desde el acceso directo ubicado en el escritorio.

Para poder conectar con el receptor y debido la actual conexión con IP publica dinámica, se ha utilizado un servicio de DDNS como No-IP. De esta forma, accederemos al receptor mediante un nombre de dominio y es este servicio, el encargado de asociar la IP publica en uso en ese mismo momento.

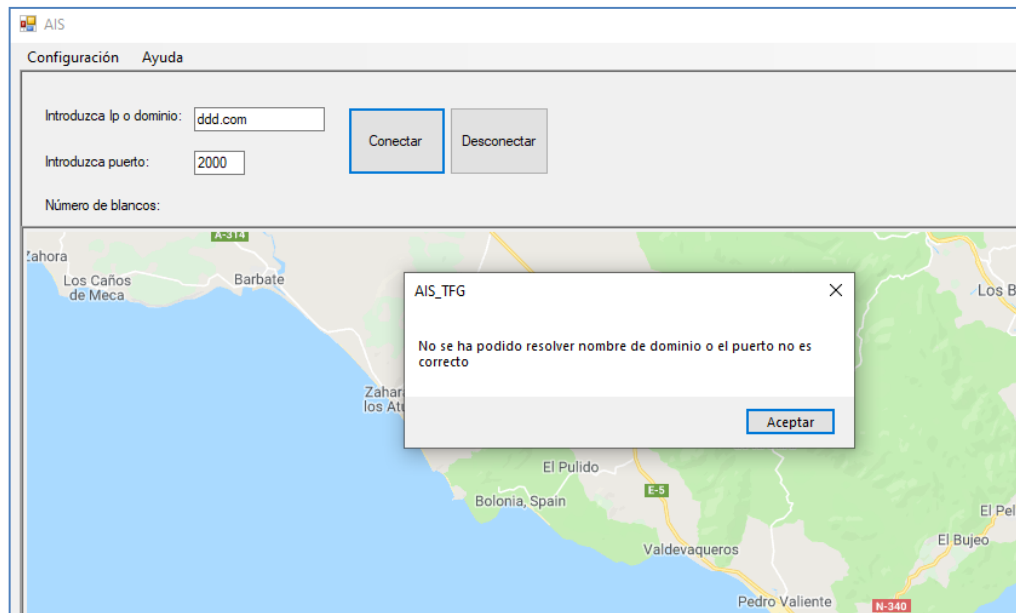
Introducimos el domino [aistfg.ddns.net](http://aistfg.ddns.net) y el puerto en el que está a la escucha el servidor Socket es **2000**. Si estuviésemos en la misma LAN que el receptor podríamos poner la dirección IP.



*Ilustración 37 Introducimos dominio/IP y puerto*

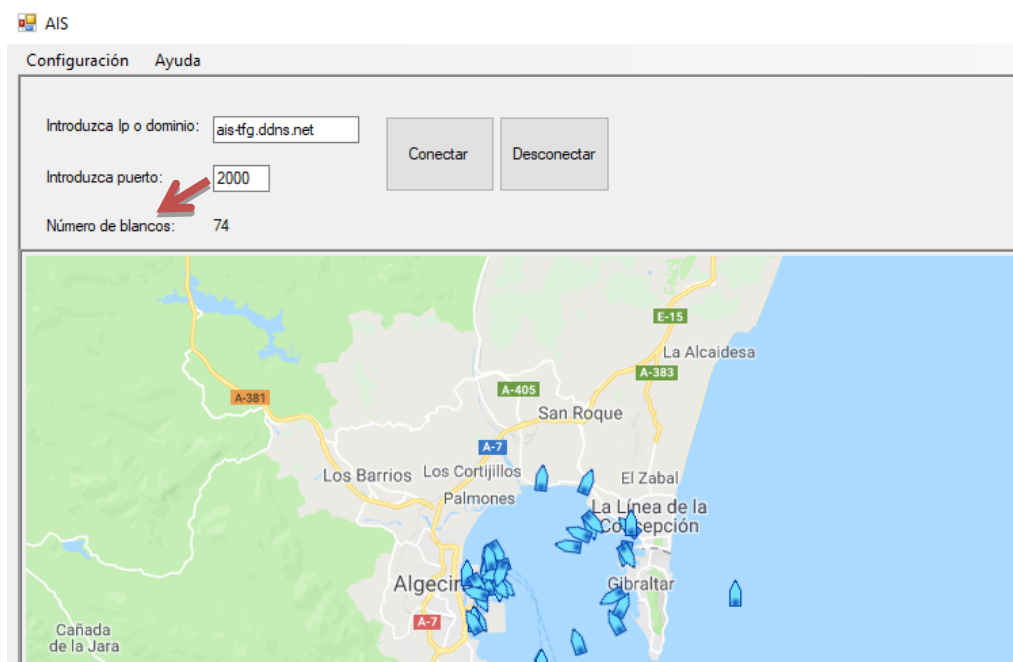
En caso de no introducir el domino/IP o puerto, la aplicación contestará mediante un mensaje descriptivo el tipo de error cometido.

En caso de no poder resolver el nombre de dominio responderá indicando que no ha sido posible resolver el nombre de dominio.



*Ilustración 38 Resolver nombre de dominio*

En el formulario principal podremos ver el número de blancos dentro de cobertura del receptor. Además, tendremos la opción de “conectar” y “desconectar” la conexión Socket

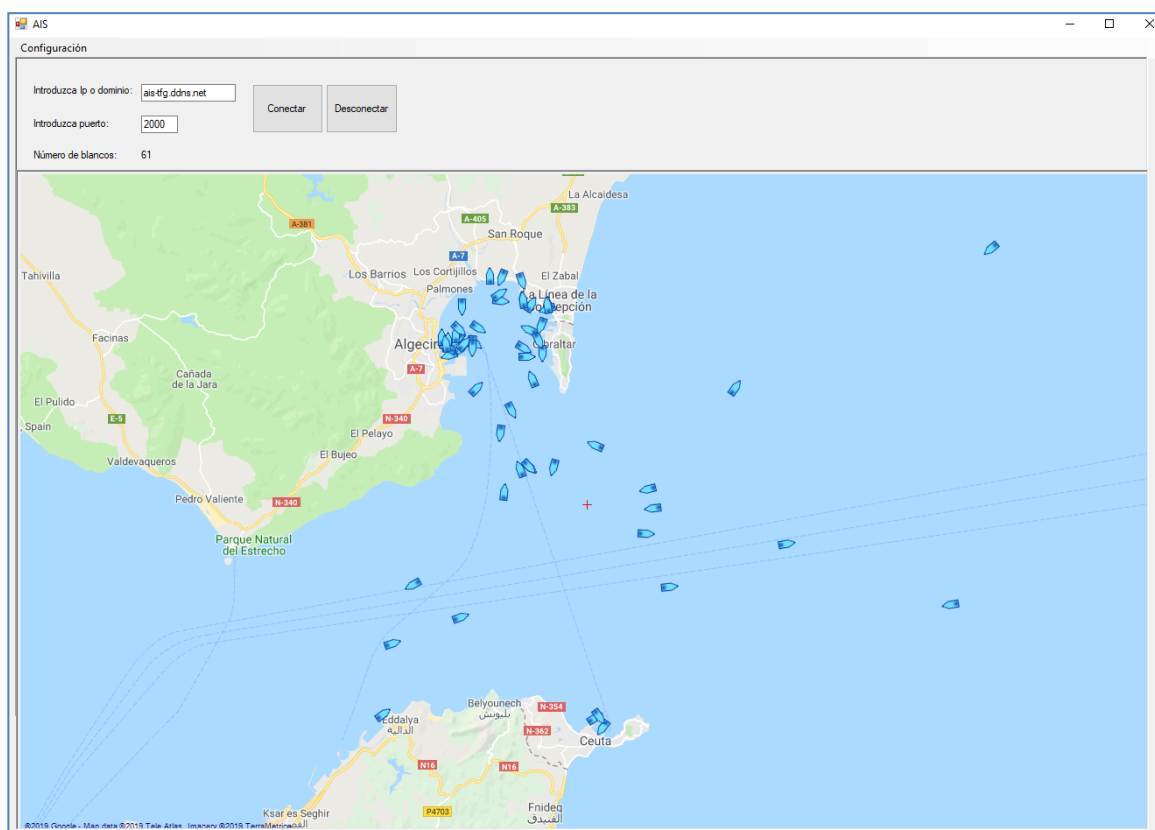


*Ilustración 39 Numero de blancos, Conectar y Desconectar Socket*

## Operación

La comprobación en la operativa se basará en el correcto funcionamiento de la aplicación. Se comprobará la recepción de los campos transmitidos por los buques cada cierto tiempo. Por último, podemos comprobar el borrado de los marcadores cuanto ha transcurrido un tiempo desde que el buque sale del alcance del receptor.

Una vez en funcionamiento podemos ver el sistema en operación visualizando el tráfico marítimo en tiempo real.



*Ilustración 40 Sistema en operación*

Se puede observar cómo se van moviendo y la dirección que lleva conforme se van recibiendo nuevas posiciones.

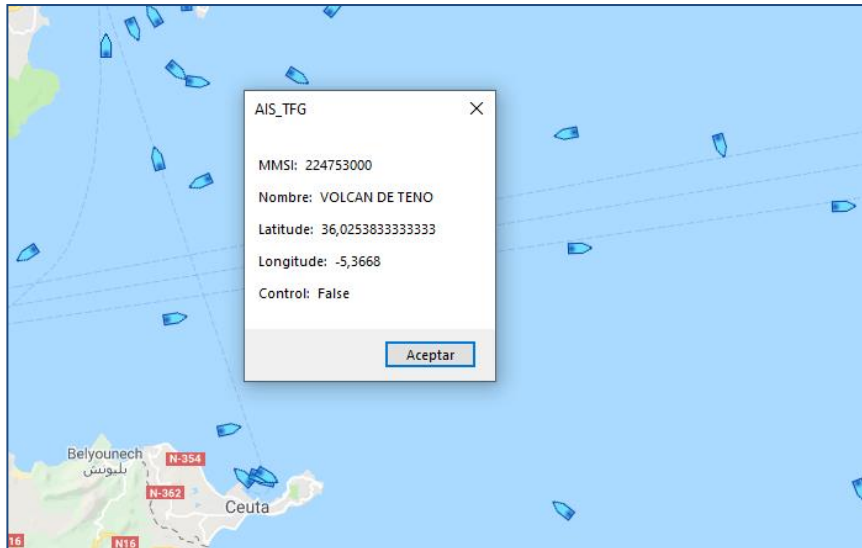
Podemos realizar las siguientes interacciones sobre el mapa:

- Podemos hacer Zoom + y Zoom – moviendo el scroll del ratón.
- Para mover el mapa haremos clic sobre él y dejándolo pulsado moveremos el ratón.

Los tipos de datos recibidos en el alcance de este proyecto son: datos dinámicos y datos estáticos.

*Datos estáticos.* Para comprobar hacemos clic sobre un icono determinado y aparecerá la información correspondiente a dicho buque. Entre la información podemos ver que, transcurrido el tiempo necesario en el que el buque transmite los campos estáticos (6 minutos), aparece el nombre.

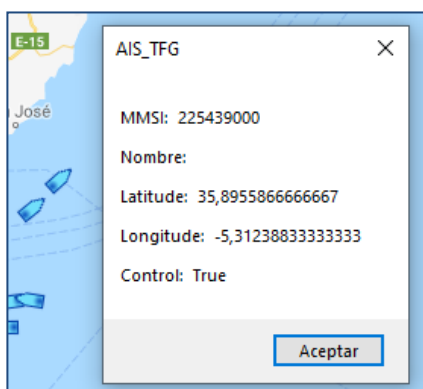
*Datos dinámicos.* Estos datos como MMSI, Latitud y longitud también se pueden ver mediante la ventana de información.



*Ilustración 41 Información del buque*

Una vez el barco se aleja de la zona de cobertura, el receptor AIS deja de recibir información con lo que, pasado un tiempo, en nuestro caso 10 minutos, se tendrán que borrar.

Para realizar esta tarea se crea un campo booleano de control que se inicializa con valor false y pasado un timeout de 10 minutos se cambia el valor a true. Cada vez que se recibe una nueva posición vuelve a false y cuando se ejecuta el siguiente timeout (otros 10 minutos) se borra del arraylist los que quedaron con valor true ya que esto significa que no se ha recibido ningún dato durante el tiempo prefijado.



*Ilustración 42 Campo de control para el borrado con valor true.*

## Modo depuración

Para poder comprobar el funcionamiento y los posibles problemas del sistema, se ha implementado un modo depuración. Para entrar en el modo depuración hacemos clic en la barra de menú “Configuración” y a continuación en depurar. Nos aparece un formulario solicitando una contraseña, introducimos 12345 y haremos clic en Aceptar o presionaremos la tecla Intro.

### Visualización de sentencias

En este modo se puede visualizar las sentencias conforme se van recibiendo.

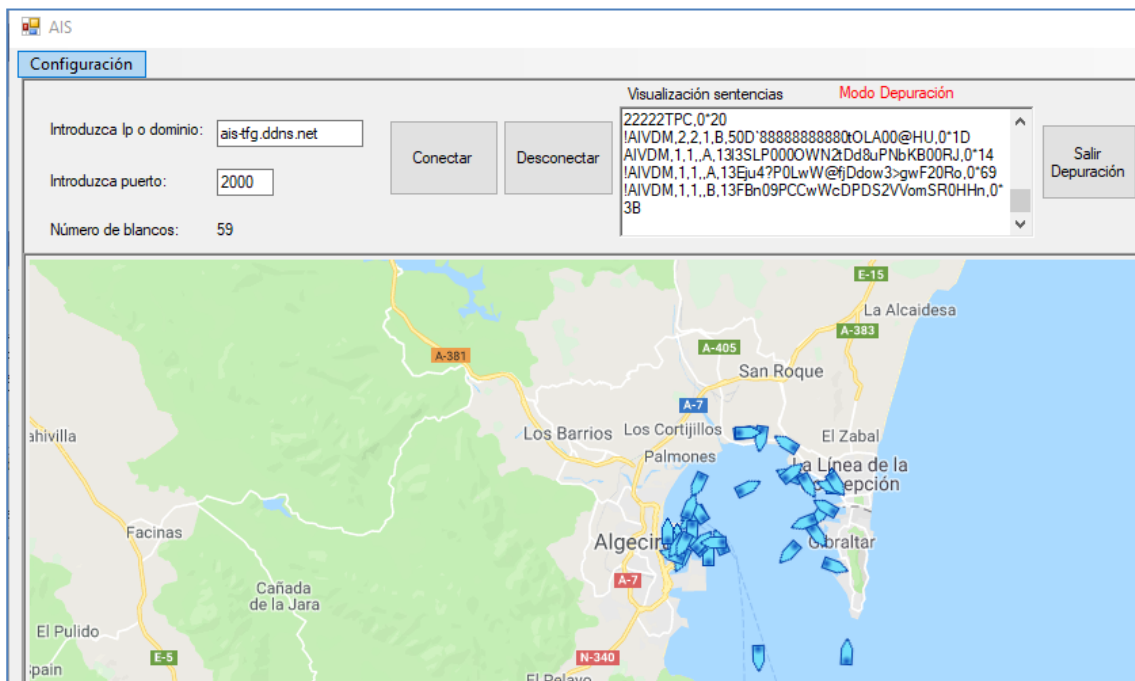


Ilustración 43 Visualización de las sentencias

Además, en este mismo modo, podemos ver una lista con los buques que actualmente se encuentran dentro de la cobertura con el receptor. Podremos actualizar la lista mediante el botón “Actualizar”.

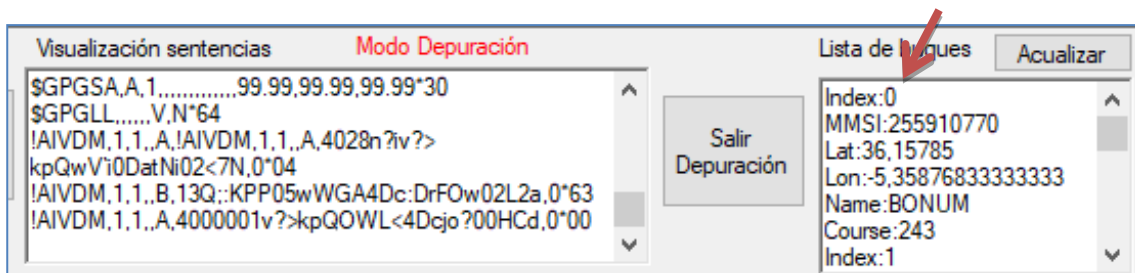


Ilustración 44 Sentencias decodificadas

## Borrado

Pasados los 10 primeros minutos se pone el campo de control con valor True, si se recibe una nueva posición, se pasa de nuevo el a False. Los que se quedaron con valor true se borrarán en los siguientes 10 minutos. Por tanto, se borrarán los primeros marcadores, en el caso de que existan buques que salen de la cobertura del receptor, tras 20 minutos una vez ejecutada la aplicación.

Podemos ver la lista de los buques que quedaron con el campo de control en true y una vez pasado de nuevo el timeout se borrarán.

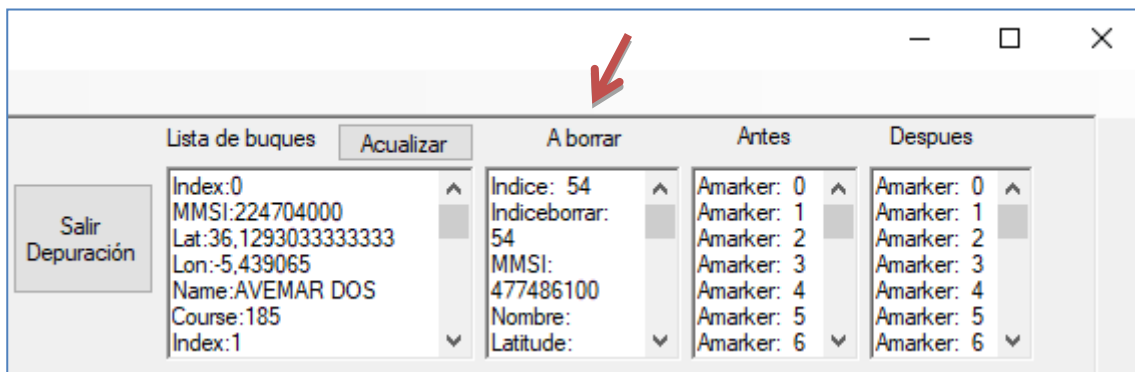


Ilustración 45 Lista con los buques a borrar

Una vez borrado los buques, se imprime la lista con el número de marcadores que había y hay en el array de marcadores.



Ilustración 46 Lista con marcadores antes y después de borrar

En la lista “Antes” y “Después”, podemos ver que había 68 marcadores y después del borrado quedaron 66.

## 7.Conclusiones

Terminado el trabajo, pienso que ha sido una experiencia muy beneficiosa para complementar mi formación. He ido viendo como a partir de la planificación, análisis y diseño se han ido alcanzando los hitos marcados.

Respecto al trabajo de la comunicación con el receptor AIS, me ha resultado muy fructuoso trabajar con Sockets implementados mediante protocolos orientados a conexión como TCP. En aplicaciones de este tipo que representan situaciones de tráfico marítimo a tiempo real, es fundamental garantizar la transmisión de los datos sin errores y en el mismo orden en que se van recibiendo.

Investigar sobre el sistema AIS, me ha permitido la posibilidad de trabajar con protocolos como NMEA, muy utilizado en equipos electrónicos marinos el cual, me ha proporcionado un mayor conocimiento y soltura a la hora de programar operaciones a nivel de bits.

Otro tema muy importante ha sido trabajar con herramientas de posicionamiento geográfico. Cada vez más, veo aplicaciones que utilizan este tipo de soluciones para representar de manera visual la información que proviene de situaciones reales. Ya que la aplicación manejaba un gran número de marcadores moviéndose dinámicamente sobre el mapa, ha sido muy enriquecedor encontrar soluciones que representaran de manera natural los movimientos de un buque según se iban recibiendo nuevas posiciones.

Creo que la aplicación ha conseguido satisfacer los objetivos marcados. Pienso que el tiempo ha sido muy ajustado y fue fundamental la organización del mismo para ir cumpliendo con las tareas propuestas en la planificación.

Una vez puesta en marcha la aplicación, se me iban ocurriendo posibles mejoras como, por ejemplo: la diferenciación por colores según la tipología de los buques o la posibilidad de crear ciertas alarmas. Me gustaría seguir trabajando en esta línea e incluir todas estas funcionalidades como posibles mejoras.



## 8. Glosario

<b>AIS</b>	(Automatic identification system). Sistema de Identificación Automática. Permite a los buques transmitir su posición e información relevantes con el fin de mejorar la seguridad marítima.
<b>NMEA 0183</b>	es una especificación combinada eléctrica y de datos entre aparatos electrónicos marinos.
<b>AIVDM</b>	Sentencia recibidas procedentes de otros buques.
<b>AIVDO</b>	Sentencia con información del propio buque.
<b>TDMA</b>	(Time Division Multiple Access o TDMA). Acceso Múltiple por División de Tiempo es una técnica que permite la transmisión de señales digitales.
<b>SOLAS</b>	(Safety of Life at Sea). Convenio Internacional para la Seguridad de la Vida en el Mar. Es el tratado más importante internacional sobre la seguridad de los buques.
<b>OMI</b>	Organización Marítima Internacional. Es el organismo especializado de las Naciones Unidas responsable de la seguridad y protección de la navegación y de prevenir la contaminación del mar por los buques
<b>IALA</b>	(International Association of Lighthouse Authorities). Asociación Internacional de Señalización Marítima es una norma internacional dictada para estandarizar las características de las señalizaciones marítimas.
<b>USB</b>	(Universal serial bus) Bus Universal en Serie es un bus de comunicaciones que sigue un estándar que define los cables, conectores y protocolos usados en un bus de comunicaciones.
<b>ASCII</b>	(American Standard Code for Information Interchange). Código Estándar Estadounidense para el Intercambio de Información.
<b>RS232-RS422</b>	(Recommended Standard 232,422). Es una interfaz que designa una norma para el intercambio de datos binarios serie.
<b>ITU</b>	(International Telecommunication Union). Es el organismo especializado de las naciones unidas para las tecnologías de la información y comunicación.
<b>BIG ENDIAN</b>	Es un criterio para representar bytes. En este formato los bits van desde el más significativo al menos significativo.

<b>UTC</b>	(Universal Time Coordinated). Tiempo Universal Coordinado. Es el principal estándar de tiempo por el cual el mundo regula los relojes y el tiempo.
<b>SOCKET</b>	o conector es un punto de acceso a los servicios de comunicación bidireccional en el ámbito de transporte.
<b>VTs</b>	(Vessel traffic services). Servicio de Tráfico Marítimo. Es un Servicio para el control de tráfico de buques.
<b>DSC</b>	(Digital Selective Calling). Llamada selectiva digital. Técnica de transmisión automática de llamadas por radio.
<b>NMEA</b>	(National Marine Electronics Association). Asociación Nacional de Fabricantes de Electrónica.
<b>WGS 84</b>	(World Geodetic System 1984) es un sistema geodésico de coordenadas geográficas usado mundialmente, que permite localizar cualquier punto de la Tierra.

## 9. Bibliografía

### AIS

#### **SISTEMA DE IDENTIFICACIÓN AUTOMÁTICA**

**En el texto:** ("Sistema de Identificación Automática", n.d.)

**Bibliografía:** Sistema de Identificación Automática. Retrieved 2 October 2019, from

[https://es.wikipedia.org/wiki/Sistema\\_de\\_Identificaci%C3%B3n\\_Autom%C3%A1tica](https://es.wikipedia.org/wiki/Sistema_de_Identificaci%C3%B3n_Autom%C3%A1tica)

#### **SOLAS CHAPTER V - ANNEX 17 - AUTOMATIC IDENTIFICATION SYSTEMS (AIS)**

**En el texto:** ("Solas Chapter V - Annex 17 - Automatic Identification Systems (AIS)", n.d.)

**Bibliografía:** Solas Chapter V - Annex 17 - Automatic Identification Systems (AIS). Retrieved 6 October 2019, from

<http://solasv.mcga.gov.uk/Annexes/Annex17.htm>

### **AIS, E.**

El sistema AIS

**En el texto:** (AIS, n.d.)

**Bibliografía:** AIS, E. El sistema AIS. Retrieved 6 October 2019, from

<https://www.farodeluisu.es/p/el-sistema-ais.html>

#### **AUTOMATIC IDENTIFICATION SYSTEMS (AIS)**

**En el texto:** ("Automatic Identification Systems (AIS)", n.d.)

**Bibliografía:** Automatic Identification Systems (AIS). Retrieved 9 October 2019, from

<http://www.imo.org/es/OurWork/Safety/Navigation/Paginas/AIS.aspx>

### TDMA

#### **ACCESO MÚLTIPLE POR DIVISIÓN DE TIEMPO**

**En el texto:** ("Acceso múltiple por división de tiempo", n.d.)

**Bibliografía:** Acceso múltiple por división de tiempo. Retrieved 15 October 2019, from

[https://es.wikipedia.org/wiki/Acceso\\_m%C3%BAltiple\\_por\\_divisi%C3%B3n\\_de\\_tiempo](https://es.wikipedia.org/wiki/Acceso_m%C3%BAltiple_por_divisi%C3%B3n_de_tiempo)

## **NMEA**

### **NMEA 0183**

**En el texto:** ("NMEA 0183", n.d.)

**Bibliografía:** NMEA 0183. Retrieved 15 October 2019, from

[https://en.wikipedia.org/wiki/NMEA\\_0183](https://en.wikipedia.org/wiki/NMEA_0183)

### **SERRA, J. M.**

¿Qué es el standar NMEA? « INFORMATICA A BORDO

**En el texto:** (Serra, 2012)

**Bibliografía:** Serra, J. (2012). ¿Qué es el standar NMEA? «

INFORMATICA A BORDO. Retrieved 17 October 2019, from

<http://www.informaticaabordo.com/2011/10/%C2%BFque-es-el-standar-nmea/>

### **BENNETT, P.**

Wayback Machine

**En el texto:** (Bennett, 1997)

**Bibliografía:** Bennett, P. (1997). Wayback Machine. Retrieved 17 October 2019, from

<https://web.archive.org/web/20140215150802/http://www.kh-gps.de/nmea.faq>

## **INTERPRETACIÓN DE LAS SENTENCIAS:**

### **M.1371 TECHNICAL CHARACTERISTICS FOR AN AUTOMATIC IDENTIFICATION SYSTEM USING TIME-DIVISION MULTIPLE ACCESS IN THE VHF MARITIME MOBILE BAND**

**En el texto:** ("M.1371 Technical characteristics for an automatic identification system using time-division multiple access in the VHF maritime mobile band", 2015)

**Bibliografía:** M.1371 Technical characteristics for an automatic identification system using time-division multiple access in the VHF maritime mobile band. (2015). Retrieved 9 October 2019, from

[https://www.itu.int/dms\\_pubrec/itu-r/rec/m/R-REC-M.1371-5-201402-!!!PDF-S.pdf](https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.1371-5-201402-!!!PDF-S.pdf)

### **RAYMOND, E.**

AIVDM/AIVDO protocol decoding

**En el texto:** (Raymond, 2019)

**Bibliografía:** Raymond, E. (2019). AIVDM/AIVDO protocol decoding. Retrieved 1 November 2019, from

<https://gpsd.gitlab.io/gpsd/AIVDM.html>

## **IMO236**

**EMBANKMENT, A.**

IMO International Maritime Organization

**En el texto:** (EMBANKMENT, 2004)

**Bibliografía:** EMBANKMENT, A. (2004). IMO International Maritime

[https://www.e-navigation.nl/sites/default/files/IMO\\_SN\\_Circ236.pdf](https://www.e-navigation.nl/sites/default/files/IMO_SN_Circ236.pdf)

## **IMO289**

**EMBANKMENT, A.**

IMO International Maritime Organization

**En el texto:** (EMBANKMENT, 2010)

**Bibliografía:** EMBANKMENT, A. (2010). IMO International Maritime

Organization Retrieved 26 October 2019, from

<http://vislab-ccom.unh.edu/~schwehr/papers/2010-IMO-SN.1-Circ.289.pdf>